

```

// CViewer.java /home/nlui 4/17/02
// Lui, Nancy
// 02/12/2002
// 10/1/03: removed references to hidden lists because data cleaned up
//      (\r removed)
//
/** LIBRARIES: The names of libraries contributing to the contig
 * assemblies are displayed on the perimeter of a circle.
 *
 * One can sort the libraries (and therefore the positions of      (mysql table
done 5/2/02)
 * the contigs) by species, germplasm, developmental stage,
 * tissue, or condition.                                          (done 6/13/02)
 *
 * User can sort the features within each of these categories
 * by choosing a CUSTOM sort.                                     (8/7/02, done
9/12/02)
 *
 * The sort categories are specified under the library labels      (8/7, done
8/28/02)
 * and represented as colored bars.                                (6/19, done 6/24/02)
 * The sort data are also displayed in the middle text area.
 *
 * A series of 'pre-processing' pages allow users to specify
 * libraries they wish to view, sort category, and order
 * of sort features before the applet loads.                      (7/30/02, started
9/13, done 9/20/02)
 * This 'Select Libraries' window presents a list of library
 * characteristics (germplasm, tissue, stage, condition)
 * for reference.                                                 (done 9/23/02)
 *
 * By default (if user doesn't make a selection), the
 * libraries are sorted in tissue order.                          (7/30/02, done 9/20/02)
 *
 * User can change set of libraries midstream by using
 * the 'Set libraries' button, which brings up a window          (done 9/23/02)
 * to change the libraries viewed. This same window
 * can be used to determine library characteristics
 * (just close the window afterwards without using the
 * DONE button).
 *
 * Placing the mouse pointer over a library label will
 * cause library data to display in the status bar.              (7/30/02, done
10/15/02 [>labels outside circle])
 *
 * CONTIGS: Inside the circle are points representing the
 * assemblies, with the position of each point representing
 * the contribution of particular libraries to that contig,
 * i.e., the point is located closer to a library if that
 * library contributed relatively more ESTs to that contig.      (requested
2/19; done)
 * PROPORTIONAL: The "pull" of a library is proportional
 * to the number of ESTs it contributes to the contig.
 * EQUAL: Each contributing library has one unit of "pull."
 * EQUAL representation is the default view.                     (8/7, done 8/26/02)
 * WEIGHTED: reflects weighting by #ESTs from library in
 * contig relative to total #ESTs in library.                   (6/28, done 7/12/02)

```

- *
 - * Upon mouseover of a contig, its name is displayed in
 - * the status bar (to aid in locating and clicking on a
 - * contig). (done 6/13/02, in lieu of

5/3/02 request

- *
 - for new contig info window[!]

on mouseover)

- * When a contig is selected, either by clicking on it,
- * or entering its number in the textfield, (done 5/2/02)
- * lines are drawn from contig to contributing libraries. (3/22; done 4/30)
- * If 'proportional' representation is selected, the # of
- * ESTs from each library is displayed next to these lines. (done 5/3/02)
- * The component libraries/ESTs in the selected contig are
- * listed in the first text area. (done 5/2/02, multiple

contigs 9/12/02)

- *
 - * Multiple contigs can be highlighted by typing/pasting in a
 - * whitespace-delimited list of contigs in the same textfield. (8/7/02, \t

done 8/27/02, \s done 9/12/02)

- * The list can be cut/pasted from the list of contigs
- * in the lower text area or an outside application. (7/31/02 [Lazo: "5

dropboxes"], done 9/12/02)

- * Only the first contig in the list will have lines drawn. (9/12/02)
- *
 - * After clicking on a contig, its location can be displayed
 - * even after switching to another library representation
 - * mode. To compare proportional/equal/weighted locations,
 - * click on a contig in one mode, then enter its number
 - * in the contig textfield before/after changing modes. (5/3,6/26, finally

done 7/15)

- *
 - * A group of contigs can be selected and displayed in
 - * lower text area by dragging the
 - * mouse pointer over the selected area of contigs. (6/26/02, done

6/26/02)

- * This feature is also helpful if a point includes more
- * than one contig--if the mouse is clicked instead of
- * dragged, all the contigs at that point are displayed
- * in the text area.
- *
 - * One can highlight the contigs to which a library
 - * contributes by selecting the library name from the
 - * choice list. (textfield 6/13/02, choice list

6/28/02)

- * One can see contigs from a 2nd/3rd library. (no request, done

9/12/02)

- *
 - * A HELP frame provides information on using the applet. (6/14/02, then

revised as features added)

- *
 - * A "SHOW BEST HIT(S)" button brings up a window
 - * with the best BLAST hit for a contig. (6/19, done 6/20/02)
 - * (Code for this has to be revised if we start using
 - * contigs from other assemblies, as they have
 - * different prefixes.)

* If multiple contigs entered, multiple browser windows
* pop up with the best hit for each. (7/30/02, done
9/12/02)

*

*

* MULTITHREADING (one thread was done 7/10/2002

* to do the library pull calculations while GUI elements

* load, but removed later when pre-processing pages developed 9/2002)

*

*

* CUSTOMIZATION

* The default sort of the various categories is alphanumeric.

* The sort can be customized by selecting CUSTOM.

* Two frames will pop up so that the user can change

* the order of the sort features. (6/28 & 8/7/02, done

9/12/02)

*

* FUTURE DEVELOPMENT

* If the number of libraries associated with contigs

* exceeds 34, this program must be modified, as it

* currently can handle only 1-34 libraries.

* (#31 - #34 have been coded but NOT yet tested) (7/3/02)

* (Had to comment out much of code in calculateXY()

* because of exception (error) for having a method

* with code over 65,000 bytes; ...uncommented 9/13/02) (7/11/02)

* (Update: In coding frLibraries frame, there may be

* other than 28 libraries, so broke up calculateXY().

* Need to uncomment out calcXY_31to33() and _34()

* when we get more libraries.) (9/13/02)

*/

//*****

// TO DO:

//*****

/**10/15/02

microarray data to follow...

*/

/** 9/24/2002 Gerry to reconsider 8/7 request:

"3. OPTION Library Consolidation by class

For example, we have 28 libraries and 11 tissue classes labelled.

A consolidation option would display 11 points around the

circle (replacing library names with tissue types). The

consolidation option would allow looking specifically at the

different classes relative to each other; allowing comparisons

of species, germplasm, tissue, dev stage, and condition

directly. This could be a fallback if library lists go over

the limit (still at 28?)."

My point: this comparison already done with libraries grouped together by
sort feature.

He agreed that it was unclear where on the "spoke" of the wheel one would
place a contig

associated with 'root tip' library (or 'root tip' library as well as
'anther'

library, etc. for that matter).

*/

/**Mon 8/19/02

Gerry: see microarray explorer freeware with java source code that might be helpful for contig

viewer. 8/26/02 looked at...prima facie not useful

*/

/** 7/30/02 one remaining suggestion from lab mtg: Yong--choose leaf, root, then subtract out commonalities

(Can kind of do this by choosing leaf and root libraries, highlighting contigs from those libraries,

then seeing which contigs are in both root/leaf libs and which contigs are only in one lib.)

*/

/** 7/25/02 Gerry will give me data on pathways for contigs so that they can be displayed in

different colors; 9/24/02 He said this is NO LONGER NEEDED since can get BEST HIT

info on multiple contigs.

*/

/** 5/2/02 tried non-tomcat to see if any faster see CViewer.javausesDataClient (Frank's never done successfully) --> try again later?

*/

/** 7/16/02: Lazo: changed his mind about how to do the weighting -- will provide algorithm later

(but it isn't what I have now). 9/24/02: Lazo OK with current method

*/

/** tested CViewer.jar: 73 seconds to load applet, 51 seconds to sort
unjarred: 70 seconds to load applet, 51 seconds to sort

*/

/**
// START PROGRAM
/**

// java.lang (needed for Math) automatically imported into all programs

import java.awt.*;

import java.awt.event.*; // package needed for buttons/checkboxes

import java.applet.*;

import java.util.*; // pkg for vectors, hashtables

import java.net.*; // package need for URL class

import java.sql.SQLException; // for database queries?

import java.util.StringTokenizer; // to parse list of contigs in text field

//<applet code = "CViewer" width=950 height=580></applet>

/**

// BEGIN CLASS CViewer

/**

public class CViewer extends Applet implements ActionListener, ItemListener,
MouseListener, MouseMotionListener

{

// global variables

```

Font smallFont = new Font("SansSerif", Font.PLAIN, 9);
Font tinyFont = new Font("SansSerif", Font.PLAIN, 8);

// CViewer (String threadName) doesn't work

//debug
String err1;
String err2;
String err3;
String err9;
String err10;
//end debug

final static int START_X=53;           // starting x,y points for circle
final static int START_Y=50;
final static int CIRCLE_RADIUS_INT=241; // distance from center of circle to
perimeter
final static double CIRCLE_RADIUS=241; // distance from center of circle to
perimeter

final static int x_ADJUSTMENT=52; // used to move libraryNames outside of
circle
final static int y_ADJUSTMENT=11;

final static int MAX_NO_COLORS =15;
final static int MAX_LENGTH=14; // max sort feature string length

final static Color BARK =          new Color(70,0,0);      // library color
final static Color BERRY =         new Color(255,60,100);  // library color
final static Color BRIGHT_BLUE =  new Color(51,153,255);  // 3399ff //
library color
final static Color BRIGHT_YELLOW = new Color(255,255,51);  // ffff33 //
library color
final static Color DARK_BLUE =      new Color(51,51,153);  // 333399 //
library color
final static Color DARK_BROWN =     new Color(51,0,0);     // 330000 for text
final static Color DARK_GREEN =     new Color(0,51,0);     // 003300
final static Color DARK_ORANGE =    new Color(204,51,0);   // cc3300
final static Color DARK_RED =       new Color(153,0,0);    // 990000
final static Color DARKER_YELLOW =  new Color(255,255,180); // darker for
buttons
final static Color FUCHSIA =         new Color(250,120,250); // library
color
final static Color GRAPE =          new Color(175,0,235);  // library color
final static Color JADE =           new Color(0,140,150);  // library color
final static Color LIGHT_BROWN =    new Color(102,51,0);   // 663300
final static Color LIGHT_OLIVE =    new Color(102,102,0);  // 666600 //
library color
final static Color LIGHTER_YELLOW = new Color(255,255,225); // lighter
shade for text areas
final static Color LIME =           new Color(180,240,0);  // library color
final static Color MEDIUM_GREEN =   new Color(0,102,0);    // 006600 //
library sort area titles
final static Color ORANGE =         new Color(255,153,102); // ff9966 //
library color
final static Color PINK =           new Color(250,180,180); // library color

```

```

    final static Color SILVER =          new Color(120,120,120); // library
color
    final static Color SKY =             new Color(0,220,255);  // library color
    final static Color TEAL =            new Color(0,255,204);  // 00ffcc  //
library color
    final static Color WHITE_YELLOW =    new Color(255,255,204); // fffffc
APPLET background

    boolean initialLoad=true;            // initial interface != later
interface

    boolean showProportional=false;      // radio buttons for library
representation
    boolean showEqual=true;
    boolean showWeighted=false;

    boolean lookingForContig = false;    // contig entered in textfield
    boolean lookingForContigs = false;   // contigs entered in textfield

    boolean showContigsInLib1 = false;   // library selected from choice
list 1
    boolean showContigsInLib2 = false;   // library selected from choice
list 2
    boolean showContigsInLib3 = false;   // library selected from choice
list 3

    boolean drawRectAroundContigs = false; // mouse pressed/dragged/released
    boolean getListOfContigs = false;

    boolean allContigsFound = true;      // no errmsg if all contigs (after
2nd) in list found

    DataClient server;                  // class to get database connection

    int clickMouseX = 0;                 // x,y coordinates for mouseclick/mouseover
    int clickMouseY = 0;
    int moveMouseX = 0;
    int moveMouseY = 0;

    int pressMouseX = 0;                 // for drawing rectangle around set of contgs
    int pressMouseY = 0;
    int releaseMouseX = 0;
    int releaseMouseY = 0;

    int xCircle;                         // start points for drawing contigs & libraries
    int yCircle;                         // (used to save time doing arithmetic
calculation)

    int numberOfESTs;                    // based on estCountByLib and number of libraries
selected
    int numberOfLibraries;               // number of libraries (after user selections
made)
    int numberOfSortableLibs;            // number of libraries from mysql's sort criteria
table
    int numberOfContigs;                 // number of distinct contigs (after user
selections made)

```

```

    int origNumberOfLibraries; // # distinct libraries when database first
queried
    int origNumberOfContigs;   // # distinct contigs when database first queried

    int contigCount;           // used to populate contig x,y array

    double pSizeOfMovement;    // how far contig point moves within circle dep.
on PROPORTIONAL library contribution
    double eSizeOfMovement;    // how far contig point moves within circle dep.
on EQUAL library contribution
    double wSizeOfMovement;    // how far contig point moves dep. on WEIGHTED
library contribution
    int xPContig,yPContig;     // coordinates of PROPORTIONAL contig
    int xEContig,yEContig;     // coordinates of EQUAL contig
    int xWContig,yWContig;     // coordinates of WEIGHTED contig

    int totalLibsInContig;     // total number of libraries in contig;
    int pNumESTsPresent=0;     // (if proportional:) number of ESTs in contig
present in current library selection
    int eLibsPresent=0;        // (if equal:) number of libs in contig present in
current library selection
    int wContigSize=0;         // (if weighted: ) number of ESTs in contigs wt'd
by #ESTs in libraries

    int iContig;               // array index of selected contig;
    int numInContigList;       // number of elements in array of contigs in
requested list (textField)
    int indexRequestedLib1;    // array index of requested library (user-selected
from choice list)
    int indexRequestedLib2;    // array index of requested library (user-selected
from choice list)
    int indexRequestedLib3;    // array index of requested library (user-selected
from choice list)
    int indexMostESTs;         // array index of library with most total ESTs
    int colorCounter;          // for tracking sorted library color

    int sortFeatureCount;      // count number of unique items in
sortedStandard[] in FrameTwo

    int[] sortColorIndex;      // array to hold colorCounter # for library sort

    int[] pContribFromLibrary; // proportional "pull" of libraries in
contig aka: int pContribFromLibrary[];
                                // index "i" comparable to "i" in libraryNames[]
    int[] eContribFromLibrary; // equal "pull" of libraries in contig aka:
int eContribFromLibrary[];
    int[] wContribFromLibrary; // weighted "pull" of libraries

    int[] x;                   // array: x-coordinate of library; same as int x[];
                                // index "i" comparable to "i" in libraryNames[]
    int[] y;                   // array: y-coordinate of library

    int[] labelX;              // location of library labels (so that they appear
outside of circle)
    int[] labelY;

    int[] xCoordEContig;       // hold EQUAL x-coordinate for contigNames[i]

```

```

int[] xCoordPContig;      // PROPORTIONAL
int[] xCoordWContig;      // WEIGHTED
int[] yCoordEContig;      // hold EQUAL y-coordinate for contigNames[i]
int[] yCoordPContig;      // PROPORTIONAL
int[] yCoordWContig;      // WEIGHTED

int[] estCountByLib;      // for library representation weighted by total
ESTs from lib
int[] origESTCountByLib;  // original database query
int[] estCountBySortedLib;
int[] estCountBySpecies;
int[] estCountByGermplasm;
int[] estCountByTissue;
int[] estCountByStage;
int[] estCountByCondition;
int[] estCountByStandardSort;

int [] iContigs;          // indices for list of contigs in textField

String clickedContig = ""; // contig that user has clicked on
String clickedMode = "";  // library representation -- P/E/W -- when contig
clicked
String contigNumber = ""; // contig number entered in textbox
String contigToShow = ""; // "C_Contig"+contigNumber
String libraryRequested1 = ""; // library for which user has requested
contigs (has \r)
String libraryRequested2 = ""; // library for which user has requested
contigs (has \r)
String libraryRequested3 = ""; // library for which user has requested
contigs (has \r)
String sortAlready="custom"; // prevent resort if user requesting same
sort as immediate last sort

StringTokenizer st;       // string(s) entered in contig textfield

String[] contigNames;     // array of contig names after user selections
made
String[] origLibraryNames; // array of library names from
server.getDistinctLibraryNames()
String[] libraryNames;    // array of library names after library selections
made
String sortedLibraryNames[]; // sorted by species, etc. after user has
ordered sort features

String[] contigsInLibrary1; // if user selects a library with contigs s/he
wants highlighted
String[] contigsInLibrary2; // if user selects a library with contigs s/he
wants highlighted
String[] contigsInLibrary3; // if user selects a library with contigs s/he
wants highlighted

String[] LibInfo;         // library names from mysql table with sort
categories
String[] Species;         // all species from sql table
String[] speciesNames;    // species associated with distinct libraries for
this applet (subset of Species[])

```

```

String sortedSpecies[];    // species sorted alphanumerically
String[] libraryNamesBySpecies; // libraries sorted in sortedSpecies[] order

String[] Germplasm;
String[] germplasmNames;
String sortedGermplasm[];
String[] libraryNamesByGermplasm;

String[] Tissue;
String[] tissueNames;
String sortedTissue[];
String[] libraryNamesByTissue;

String[] Stage;    // developmental stage
String[] stageNames;
String sortedStage[];
String[] libraryNamesByStage;

String[] Condition;
String[] conditionNames;
String sortedCondition[];
String[] libraryNamesByCondition;

String[] libraryNamesByStandardSort;    // used in custom sort to hold
libraryNames before custom sort
String[] sortedStandard;    // sort features before custom sort
String[] sortedCustom;    // sort features after custom sort

String contigDataString=""; // list of component libraries/ESTs in contig
String librarySortString; // criteria by which libraries were sorted
String contigListString; // list of contigs selected by user dragging mouse

EstsOfLibrary[] estObjs;    // class defined in EstsOfLibrary.java: lib+#ESTs
in contig x

Color sortColor[];    // colors for sorted libraries

// GUI VARIABLES:

// initial window prompting user for library selection
Frame frLibraries;
java.awt.List availableLibs;    // list of available libraries
java.awt.List selectedLibs; // list of libraries selected by user
java.awt.List libInfoList; // list of data (germplasm, tissue, stage,
condition) for available libraries

Button moveAllLibs;    // Select all available libraries - middle column
Button moveLib;    // Select single / multiple library - middle column
Button removeAllLibs;    // Remove all selected libraries - right column
Button removeLib;    // Remove single library - right column
Button libsDone;    // When finished selecting libraries

// upper left corner of display
Button help;    // when button pressed, instructions in a 'help
frame' displayed
Frame frHelp;    // help frame

```

```

Label radioTitle;
Label propLabel, equalLabel, weightedLabel;
CheckboxGroup selectOption;
Checkbox Proportional, Equal, Weighted; // radio buttons to pick proportional
or equal contribution

// center top of display
Button setLibs; // when button pressed, user can choose a new set of
libraries

Label libraryTitle;
Choice whichContigsFromLib1; // choose library->all contigs from library
highlighted
Choice whichContigsFromLib2; // choose library->all contigs from library
highlighted
Choice whichContigsFromLib3; // choose library->all contigs from library
highlighted

// upper right of display
Label contigTitle;
TextField whereAreContigs; // paste tab-delimited list of contigs -->
highlight contigs
Button bestHit; // displays best BLAST hit for contig
TextArea whatsInContig; // text area to display contig's component
libraries/ESTs

// middle right of display
Label sortTitle; // need to declare 'Label' here to use setText
later
Choice libraryOrder; // choice list to order libraries by species,
germplasm, etc.
TextArea librarySortData; // text area to display criteria by which
libraries were sorted

// lower right of display
TextArea contigList; // text area to display list of selected contigs

// custom sort - these have to be defined in applet, not frame
Frame frCustom1; // custom sort frame #1 (pick sort category)
Choice libraryCustomCategory; // choice list in custom sort category pop-
up window
Button category; // in frame #1

Frame frCustom2; // custom sort frame #2 (order features within
selected sort category)
Button moveUp; // button for custom sort frame #2
Button sortDone; // " "
java.awt.List libraryCustomOrder; // list in custom sort order pop-up window

//*****

public void init() // no, can't throw exception here
//*****

{
//err1="started init: "+System.currentTimeMillis();

```

```

getKeySQLData(); // get summary data from database

sortColor = new Color[MAX_NO_COLORS]; // colors for libraries

sortColor[0] = BRIGHT_YELLOW;
sortColor[1] = ORANGE;
sortColor[2] = BERRY;
sortColor[3] = FUCHSIA;
sortColor[4] = GRAPE;
sortColor[5] = DARK_BLUE;
sortColor[6] = BRIGHT_BLUE;
sortColor[7] = SKY;
sortColor[8] = TEAL;
sortColor[9] = LIME;
sortColor[10] = JADE;
sortColor[11] = LIGHT_OLIVE;
sortColor[12] = PINK;
sortColor[13] = SILVER;
sortColor[14] = BARK;

// first, need to define custom sort frames and allocate memory
// so that they're available to frLibraries frame:

// START CUSTOM SORT window #1
frCustom1 = new CViewerCustomCategoryFrameOne("Select Sort Category"); //
to specify sort category

// choice list in pop-up window
// new Choice() must be together with addItem (in FrameOne or otherwise) to
avoid NullPointerException
libraryCustomCategory = new Choice();
libraryCustomCategory.addItem(" tissue");
libraryCustomCategory.addItem(" species");
libraryCustomCategory.addItem(" germplasm");
libraryCustomCategory.addItem(" dev_stage");
libraryCustomCategory.addItem(" condition");
// not needed to specify default, since tissue is first:
libraryCustomCategory.select(0);
libraryCustomCategory.setBounds(95,70,100,20);
libraryCustomCategory.setBackground(LIGHTER_YELLOW); // color of drop-down
default=white (which is OK, too)
libraryCustomCategory.setForeground(MEDIUM_GREEN); // color of text
// not needed: libraryCustomCategory.addItemListener(new ItemListener()

category = new Button("Go!"); // button on frame #1
category.setBackground(DARKER_YELLOW);
category.setForeground(DARK_BROWN);
category.setBounds(117,100,60,25);
category.addActionListener(new ActionListener() { public void
actionPerformed(ActionEvent ae)
{
// "synchronized" with wait() & notifyAll() doesn't work
// make sort category arrays based on sort category selection
try

```

```

    {
        // array reinitialization unnec., as all are the same size--
        numberOfLibraries
        // START 'Select category' actions
        if ( libraryCustomCategory.getSelectedItem().equals("  germplasm") )
        {
            // populate array of features sorted alphanumerically

System.arraycopy(sortedGermplasm, 0, sortedStandard, 0, numberOfLibraries);

            // populate array of library names in same order as sortedGermplasm

System.arraycopy(libraryNamesByGermplasm, 0, libraryNamesByStandardSort, 0, numberOf
Libraries);

            // populate array of EST counts in same order as sortedGermplasm

System.arraycopy(estCountByGermplasm, 0, estCountByStandardSort, 0, numberOfLibrarie
s);
        }
        else if ( libraryCustomCategory.getSelectedItem().equals("
dev_stage"))
        {
            System.arraycopy(sortedStage, 0, sortedStandard, 0, numberOfLibraries);

System.arraycopy(libraryNamesByStage, 0, libraryNamesByStandardSort, 0, numberOfLibr
aries);

System.arraycopy(estCountByStage, 0, estCountByStandardSort, 0, numberOfLibraries);
        }
        else if ( libraryCustomCategory.getSelectedItem().equals("
condition"))
        {

System.arraycopy(sortedCondition, 0, sortedStandard, 0, numberOfLibraries);

System.arraycopy(libraryNamesByCondition, 0, libraryNamesByStandardSort, 0, numberOf
Libraries);

System.arraycopy(estCountByCondition, 0, estCountByStandardSort, 0, numberOfLibrarie
s);
        }
        else if ( libraryCustomCategory.getSelectedItem().equals("  species"))
        {
            System.arraycopy(sortedSpecies, 0, sortedStandard, 0, numberOfLibraries);

System.arraycopy(libraryNamesBySpecies, 0, libraryNamesByStandardSort, 0, numberOfLi
braries);

System.arraycopy(estCountBySpecies, 0, estCountByStandardSort, 0, numberOfLibraries)
;
        }
        else // default of "tissue": if (
libraryCustomCategory.getSelectedItem().equals("  tissue"))
        {
            System.arraycopy(sortedTissue, 0, sortedStandard, 0, numberOfLibraries);

```

```
System.arraycopy(libraryNamesByTissue,0,libraryNamesByStandardSort,0,numberOfLib  
raries);
```

```
System.arraycopy(estCountByTissue,0,estCountByStandardSort,0,numberOfLibraries);  
}
```

```
    // set up choice list for frCustom2, which lets user specify order of  
sort features
```

```
    libraryCustomOrder.removeAll(); // reset list for next  
CUSTOM sort
```

```
    libraryCustomOrder.add(""+sortedStandard[0]); // add first item from  
array
```

```
    sortFeatureCount=1;  
    for (int i=1; i < numberOfLibraries; i++) // add features to list  
for FrameTwo
```

```
    {  
        if ( !sortedStandard[i].equals(sortedStandard[i-1]) ) // non-  
redundant
```

```
        {  
            sortFeatureCount++;  
            libraryCustomOrder.add(""+sortedStandard[i]);  
        }  
    } // end for
```

```
    // close this window  
frCustom1.setVisible(false);
```

```
    if ( sortFeatureCount > 1 ) // bring up another frame to change order of  
features in sort category
```

```
    {  
        frCustom2.setSize(475,370);  
        frCustom2.setLocation(435,350);  
        frCustom2.add(libraryCustomOrder);  
        frCustom2.add(moveUp);  
        frCustom2.add(sortDone);  
        frCustom2.setVisible(true);  
    }
```

```
    else // only one sort feature --> no need to choose order using  
frCustom2
```

```
    {  
        // change visible selection in libraryOrder choice list so user sees  
current sort category
```

```
        if ( libraryCustomCategory.getSelectedItem().equals(" germplasm") )  
        {  
            libraryOrder.select(" germplasm"); // change selection displayed in  
choice list
```

```
            sortAlready="germplasm";  
        }
```

```
        else if ( libraryCustomCategory.getSelectedItem().equals("dev_stage"))
```

```
        {  
            libraryOrder.select(" dev_stage");  
            sortAlready="dev_stage";  
        }
```

```
        else if ( libraryCustomCategory.getSelectedItem().equals("condition"))
```

```

        {
            libraryOrder.select(" condition");
            sortAlready="condition";
        }
        else if ( libraryCustomCategory.getSelectedItem().equals("
species"))
        {
            libraryOrder.select(" species");
            sortAlready="species";
        }
        else // default of "tissue": if (
libraryCustomCategory.getSelectedItem().equals(" tissue"))
        {
            libraryOrder.select(" tissue");
            sortAlready="tissue";
        }
    }

    // don't bring up frame to change order; go straight to
fillSortedCustom()
        fillSortedCustom(); // populate array of custom-sorted
features
    } // end if sortFeatureCount
    }
    catch (Exception e)
    {
        err1=err1+"frCustom1CategorySelect error: "+e;
    }
    // END 'Select category' actions
    } // end actionPerformed
    }); // end ActionListener

    // back to init() and CUSTOM SORT frame #1 definition:
    frCustom1.setSize(300,165); // required to specify size of
frame
    frCustom1.setLocation(665,380); // near original libraryOrder choice
list
    frCustom1.add(libraryCustomCategory); // has to be in applet (can't add
inside Frame definition)
    frCustom1.add(category);
    // END CUSTOM SORT window #1

    // START CUSTOM SORT window #2
    // can't put this in itemStateChanged for frCustom1's choice list:
    frCustom2 = new CViewerCustomFeatureFrameTwo("Select Sort Order"); // to
specify sort order

    // LIST OF FEATURES IN SELECTED SORT CATEGORY
    libraryCustomOrder = new java.awt.List(12,false); // display 12 items, only
one selection allowed
    libraryCustomOrder.setBounds(25,80,200,240); // couldn't use
sortFeatureCount*12 for height
    libraryCustomOrder.setBackground(LIGHTER_YELLOW); // color of drop-down
default=white (which is OK, too)
    libraryCustomOrder.setForeground(MEDIUM_GREEN); // color of text
    // unnecessary: libraryCustomOrder.addItemListener

```

```

// BUTTON TO CHANGE ORDER OF SORT FEATURES
moveUp = new Button("MOVE UP");
moveUp.setBackground(DARKER_YELLOW);
moveUp.setForeground(DARK_BROWN);
moveUp.setBounds(250,110,70,30);
moveUp.addActionListener(new ActionListener() { public void
actionPerformed(ActionEvent ae)
{
    if ( libraryCustomOrder.getSelectedIndex() != 0 ) // item selected
isn't the first item
    {
        int indexNum=libraryCustomOrder.getSelectedIndex();

        // put preceding item in temporary holding place
        String temp=libraryCustomOrder.getItem(indexNum-1);

        // insert item to be moved up

libraryCustomOrder.replaceItem(libraryCustomOrder.getSelectedItem(),indexNum-1);

        // move preceding item down one row in list
        libraryCustomOrder.replaceItem(temp,indexNum);

        // keep item selected to facilitate next 'Move Up'
        libraryCustomOrder.select(indexNum-1);
    } // end if
                                } // end actionPerformed
                                }); // end ActionListener

// DONE button pushed when features have been placed in desired order
sortDone = new Button("DONE");
sortDone.setBackground(DARKER_YELLOW);
sortDone.setForeground(DARK_BROWN);
sortDone.setBounds(250,220,70,30);
sortDone.addActionListener(new ActionListener() { public void
actionPerformed(ActionEvent ae)
{
    // synchronized (this) with wait() and notifyAll() doesn't work
    frCustom2.setVisible(false); // close the second window
    sortAlready="custom"; // for proper sort feature
labels around circle
        libraryOrder.select(" CUSTOM"); // change selection
displayed in choice list
        fillSortedCustom(); // populate array of custom-
sorted features
                                } // end actionPerformed()
                                }); // end ActionListener()

// END CUSTOM SORT window #2

// START GUI for window to choose libraries
frLibraries = new CViewerChooseLibFrame("Select Libraries");

// reference list-static (no selections)
libInfoList = new java.awt.List(8);
libInfoList.setBounds(25,100,380,435);
libInfoList.setForeground(Color.black);

```

```

// add info to guide user in selection
// from arrays populated from mysql table that includes non-contigged
libraries
for (int i=0; i< numberOfSortableLibs; i++)
{
    libInfoList.add(LibInfo[i].trim()+" "
        +Germplasm[i].trim()+" "
        +Tissue[i].trim()+" "
        +Stage[i].trim()+" "
        +Condition[i].trim());
} // end for // trim() to strip any special
characters

// list of available libraries
availableLibs = new java.awt.List(8,true); // list of available
libraries; T:mult select OK
availableLibs.setBounds(425,100,90,435);
availableLibs.setBackground(LIGHTER_YELLOW); // color of button
availableLibs.setForeground(DARK_BROWN); // color of text

// add libraries
for (int i=0; i< origNumberOfLibraries; i++)
{
    availableLibs.add(origLibraryNames[i].trim()); // take out \r for
display
}

// list for user-selected libraries
selectedLibs = new java.awt.List(8,false); // list of libraries
selected by user; F:single selection only
selectedLibs.setBounds(660,100,90,435);
selectedLibs.setBackground(LIGHTER_YELLOW);
selectedLibs.setForeground(MEDIUM_GREEN);

moveAllLibs = new Button("Move All Libs"); // Select all available
libraries - middle column
moveAllLibs.setBackground(DARKER_YELLOW);
moveAllLibs.setForeground(MEDIUM_GREEN);
moveAllLibs.setBounds(535,120,110,30);
moveAllLibs.addActionListener(new ActionListener() { public void
actionPerformed(ActionEvent ae)
{
    // capture current # of items in available list
    int numAvailable = availableLibs.getItemCount();

    for (int i=0; i<numAvailable; i++)
    {
        selectedLibs.add(availableLibs.getItem(0));
        availableLibs.remove(0);
    }

} // end actionPerformed()
}); // end ActionListener()

moveLib = new Button("Move Libs -->"); // Select single library -
middle column
moveLib.setBackground(DARKER_YELLOW);

```

```

        moveLib.setForeground(MEDIUM_GREEN);
        moveLib.setBounds(535,170,110,30);
        moveLib.addActionListener(new ActionListener() { public void
actionPerformed(ActionEvent ae)
        {
            int numSelected = availableLibs.getSelectedItems().length;    //
original # of selected items

            for (int i=0; i< numSelected; i++)
            {
                selectedLibs.add(availableLibs.getSelectedItems()[0]);    // add items
to selection
                availableLibs.remove(availableLibs.getSelectedItems()[0]);    //
remove from first list
            }

        } // end actionPerformed()
    }); // end ActionListener()

    removeAllLibs = new Button("Remove All"); // Remove all selected libraries
- right column
    removeAllLibs.setBackground(DARKER_YELLOW);
    removeAllLibs.setForeground(Color.red);
    removeAllLibs.setBounds(775,120,90,30);
    removeAllLibs.addActionListener(new ActionListener() { public void
actionPerformed(ActionEvent ae)
    {
        int numSelected = selectedLibs.getItemCount(); // total number in
'SELECTED' list

        for (int i=0; i< numSelected; i++)
        {
            availableLibs.add(selectedLibs.getItem(0));
            selectedLibs.remove(0);
        }

    } // end actionPerformed()
    }); // end ActionListener()

    removeLib = new Button("Remove Lib"); // Remove single library - right
removeLib.setBackground(DARKER_YELLOW);
removeLib.setForeground(Color.red);
removeLib.setBounds(775,170,90,30);
removeLib.addActionListener(new ActionListener() { public void
actionPerformed(ActionEvent ae)
    {
        if ( selectedLibs.getSelectedIndex() > -1 )    // don't do anything
unless something selected
        {
            availableLibs.add(selectedLibs.getSelectedItem());
            selectedLibs.remove(selectedLibs.getSelectedIndex());
        }

    } // end actionPerformed()
    }); // end ActionListener()

    // Choice lists for user to select library whose contigs user wants to
highlight

```

```

    // Need to define here, outside of ActionListener for libsDone, where it is
first
    // used, because these lists are also used later when 'setLibs' button is
clicked.
    whichContigsFromLib1=new Choice();
    whichContigsFromLib1.setBounds(500,18,103,18);
    whichContigsFromLib1.setBackground(LIGHTER_YELLOW);
    whichContigsFromLib1.setForeground(DARK_BROWN);

    whichContigsFromLib2=new Choice();
    whichContigsFromLib2.setBounds(500,38,103,18);
    whichContigsFromLib2.setBackground(LIGHTER_YELLOW);
    whichContigsFromLib2.setForeground(DARK_BROWN);

    whichContigsFromLib3=new Choice();
    whichContigsFromLib3.setBounds(500,58,103,18);
    whichContigsFromLib3.setBackground(LIGHTER_YELLOW);
    whichContigsFromLib3.setForeground(DARK_BROWN);

    libsDone = new Button("DONE");          // button clicked when finished
selecting libraries
    libsDone.setBackground(LIGHTER_YELLOW);
    libsDone.setForeground(MEDIUM_GREEN);
    libsDone.setBounds(785,450,70,30);
    libsDone.addActionListener(new ActionListener() { public void
actionPerformed(ActionEvent ae)
    {
        // START libsDone processing
        try
        {
            libsDone.setLabel("Wait...");    // alert user to wait while library
selections processed

            // set the number of libraries for applet
            numberOfLibraries = selectedLibs.getItemCount();    // # items in
selection list

            if ( numberOfLibraries > 0 )
            {
                libraryNames = selectedLibs.getItems();          // bring over the
libs (w/ \r) selected
            }
            else // user didn't specify any libraries ---> get all of them!
            {
                libraryNames = new String[origNumberOfLibraries]; // need this memory
allocation

                System.arraycopy(origLibraryNames,0,libraryNames,0,origNumberOfLibraries);
                numberOfLibraries = origNumberOfLibraries;
            } // end if

            // rearrange estCounts to match order in libraryNames
            estCountByLib=          new int[numberOfLibraries];

            for (int i=0; i<origNumberOfLibraries; i++)
            {

```

```

        for (int j=0; j<numberOfLibraries; j++)
        {
            if ( libraryNames[j].equals(origLibraryNames[i]) )
            {
                estCountByLib[j]=origESTCountByLib[i];
            }
        } // end for j
    } // end for i

    //need to define these below because makeSortedLibraries() uses
speciesNames, etc.
    speciesNames =          new String[numberOfLibraries];          // sort
criteria for libraries with contigs
    germplasmNames =          new String[numberOfLibraries];
    tissueNames =              new String[numberOfLibraries];
    stageNames =              new String[numberOfLibraries];
    conditionNames =          new String[numberOfLibraries];

    libraryNamesBySpecies =   new String[numberOfLibraries]; // library
names in sortedSpecies(?) order
    libraryNamesByGermplasm = new String[numberOfLibraries];
    libraryNamesByTissue =    new String[numberOfLibraries];
    libraryNamesByStage =     new String[numberOfLibraries];
    libraryNamesByCondition = new String[numberOfLibraries];
    libraryNamesByStandardSort= new String[numberOfLibraries];

    estCountBySpecies =       new int[numberOfLibraries];    // # ESTs in
library in sortedSpecies(?) order
    estCountByGermplasm =     new int[numberOfLibraries];
    estCountByTissue =        new int[numberOfLibraries];
    estCountByStage =         new int[numberOfLibraries];
    estCountByCondition =     new int[numberOfLibraries];
    estCountByStandardSort=   new int[numberOfLibraries];

    sortedSpecies =           new String[numberOfLibraries];    //
required for arraycopy() to work
    sortedGermplasm =         new String[numberOfLibraries];
    sortedTissue =            new String[numberOfLibraries];
    sortedStage =             new String[numberOfLibraries];
    sortedCondition =         new String[numberOfLibraries];
    sortedStandard =          new String[numberOfLibraries];    //
alphanumeric sort of features in selected category
    sortedCustom =           new String[numberOfLibraries];    // user-
specified sort "

    sortedLibraryNames = new String[numberOfLibraries];          // required
for sortBySpecies() etc.
    estCountBySortedLib =    new int[numberOfLibraries];

    sortColorIndex = new int[numberOfLibraries]; // sortColor array index for
library

    // variables needed for drawLibraries()
    x = new int[numberOfLibraries]; // allocate memory for library x,y
coordinates
    y = new int[numberOfLibraries];
    labelX= new int[numberOfLibraries];

```

```

        labelY= new int[numberOfLibraries];

        pContribFromLibrary = new int[numberOfLibraries]; // PROPORTIONAL pull
of each library
        eContribFromLibrary = new int[numberOfLibraries]; // EQUAL pull from
each library
        wContribFromLibrary = new int[numberOfLibraries]; // WEIGHTED pull from
each library


        // Update 'Show contigs in library' choice lists
        // LIBRARY #1
        whichContigsFromLib1.removeAll();
        whichContigsFromLib1.add("          None          "); // extra spaces nec
for UNIX menu width
        for (int i=0; i<numberOfLibraries; i++)
        {
//          whichContigsFromLib1.addItem(libraryNames[i].trim()); // based on
libraryNames loaded in getKeySQLData()
          whichContigsFromLib1.addItem(libraryNames[i]); // based on
libraryNames loaded in getKeySQLData()
        }
        whichContigsFromLib1.select(libraryRequested1.trim()); // retain
selection between lib sorts
        if ( whichContigsFromLib1.getSelectedIndex() > -1 ) // if
something selected
        {
            showContigsFromLibrary1();
        }
        add(whichContigsFromLib1);
        whichContigsFromLib1.addItemListener(new ItemListener() { public void
itemStateChanged(ItemEvent ie)
{
showContigsFromLibrary1(); }
});

        // LIBRARY #2
        whichContigsFromLib2.removeAll();
        whichContigsFromLib2.addItem("          None          ");
        for (int i=0; i<numberOfLibraries; i++)
        {
//          whichContigsFromLib2.addItem(libraryNames[i].trim());
          whichContigsFromLib2.addItem(libraryNames[i]);
        }
        whichContigsFromLib2.select(libraryRequested2.trim());
        if ( whichContigsFromLib2.getSelectedIndex() > -1 )
        {
            showContigsFromLibrary2();
        }
        add(whichContigsFromLib2);
        whichContigsFromLib2.addItemListener(new ItemListener() { public void
itemStateChanged(ItemEvent ie)
{
showContigsFromLibrary2(); }
});

        // LIBRARY #3
        whichContigsFromLib3.removeAll();

```

```

        whichContigsFromLib3.addItem("          None          "); // extra spaces
nec for UNIX menu width
        for (int i=0; i<numberOfLibraries; i++)
        {
            whichContigsFromLib3.addItem(libraryNames[i]); // based on
libraryNames loaded in getKeySQLData()
        }
        whichContigsFromLib3.select(libraryRequested3.trim());
        if ( whichContigsFromLib3.getSelectedIndex() > -1 ) // something
selected
        {
            showContigsFromLibrary3();
        }
        add(whichContigsFromLib3);
        whichContigsFromLib3.addItemListener(new ItemListener() { public void
itemStateChanged(ItemEvent ie)
                                                    { showContigsFromLibrary3();
}
                                                    });

        // need to get subset of contigs (with ESTs from at least one of the libs
chosen)
        Vector contigHolder =
            new Vector(origNumberOfContigs,500); // temporary holding
place; increment=500;

        for (int i=0; i < numberOfLibraries; i++)
        {
            String[] contigsFromSelectedLib =
                getContigsForLibrary(libraryNames[i]); // query db->array of
contigs returned

            for (int j=0; j < contigsFromSelectedLib.length; j++)
            {
                if ( !contigHolder.contains(contigsFromSelectedLib[j]) ) // avoid
duplicates
                {
                    contigHolder.addElement(contigsFromSelectedLib[j]); // put into
vector
                }
            } // end for j
        } // end for i

        // convert vector to array contigNames
        // need "new String[contigHolder.size()]" as argument per java.sun.com
        contigNames = (String[])contigHolder.toArray(new
String[contigHolder.size()]); // cast is needed to convert Vector to array

        numberOfContigs = contigNames.length; // number of contigs with ESTs
from libraries selected

        // if not the initial library sort, highlight contigs already specified
in textField using newly-sorted libs
        if ( !initialLoad )
        {
            doWhereAreContigsProcess();
        }

```

```

        // needs to be here so that arrays initialized for calculateXY():
        xCoordPContig = new int[numberOfContigs];    // used to register x,y
coordinates of PROPORTIONAL contigs
        yCoordPContig = new int[numberOfContigs];
        xCoordEContig = new int[numberOfContigs];    // used to register x,y
coordinates of EQUAL contigs
        yCoordEContig = new int[numberOfContigs];
        xCoordWContig = new int[numberOfContigs];    // used to register x,y
coordinates of WEIGHTED contigs
        yCoordWContig = new int[numberOfContigs];

        contigHolder.removeAllElements(); // removes all elements and size=0;
same as contigHolder.clear();
        frLibraries.setVisible(false);    // close this window

        makeSortedLibraries();    // populate arrays containing sort
categories
        frCustom1.setVisible(true);    // bring up 'Sort Category' window
    }
    catch (Exception e)
    {
        err1=err1+"frLibrariesSelected error: "+e;
    }
    // END libsDone processing
                                } // end actionPerformed()
                                }); // end ActionListener()

    // back to rest of specs for frLibraries window:
    // set specs for library selection window:
    frLibraries.setSize(920,575);    // width, height
    frLibraries.setLocation(5,4);
    frLibraries.add(libInfoList);
    frLibraries.add(availableLibs);
    frLibraries.add(selectedLibs);
    frLibraries.add(moveAllLibs);
    frLibraries.add(moveLib);
    frLibraries.add(removeAllLibs);
    frLibraries.add(removeLib);
    frLibraries.add(libsDone);
    // moved to end of init(): frLibraries.setVisible(true);

    // END of GUI for 'Choose Libraries' window

    // START GUI for elements that do not depend on selections in three frames

    xCircle = START_X + CIRCLE_RADIUS_INT;    // to cut down on # of
calculations
    yCircle = START_Y + CIRCLE_RADIUS_INT;    //    in calculateXY()

    addMouseListener(this);    // register applet as mouse event
listener
    addMouseMotionListener(this);

    setLayout(null);    // kill layout manager --> manually-set
layout

```

```

setBackground(WHITE_YELLOW);
setFont(smallFont); // in paint(), g.setFont(smallFont);

// HELP
help = new Button("HELP"); // brings up help box
help.setBackground(DARKER_YELLOW);
help.setForeground(DARK_BROWN); // don't like how DARK_ORANGE looks
help.setBounds(150,10,60,20);
add(help);
help.addActionListener(new ActionListener() { public void
actionPerformed(ActionEvent ae)
{
frHelp.setVisible(true); //
user can close manually
}
});

// help window
frHelp = new CViewerHelpFrame("Help Frame Window");
frHelp.setSize(570,590); // pixels wide, pixels high

// LIBRARY REPRESENTATION RADIO BUTTONS
radioTitle = new Label("Representation", Label.CENTER);
radioTitle.setBounds(6,37,82,11);
radioTitle.setBackground(DARKER_YELLOW);
radioTitle.setForeground(DARK_BROWN);
add(radioTitle);

selectOption = new CheckboxGroup(); // two radio buttons

Proportional=new Checkbox("",false,selectOption); // buttons w/o
labels; no default selection
Proportional.setBounds(7,52,13,13); // button location
Proportional.setBackground(DARKER_YELLOW);
add(Proportional);
Proportional.addItemListener(this);

propLabel = new Label ("Proportional", Label.LEFT); // set label
manually
propLabel.setBackground(DARKER_YELLOW); // no boundary around
label
propLabel.setForeground(LIGHT_OLIVE);
propLabel.setBounds(23,53,65,12);
add(propLabel);

Equal=new Checkbox("",true,selectOption); // set as default per Lazo
request 8/7/02
Equal.setBounds(7,67,13,13);
Equal.setBackground(DARKER_YELLOW);
add(Equal);
Equal.addItemListener(this);

equalLabel = new Label ("Equal", Label.LEFT);
equalLabel.setBackground(DARKER_YELLOW);
equalLabel.setForeground(DARK_RED);

```

```

equalLabel.setBounds(23,67,60,12);
add(equalLabel);

Weighted=new Checkbox("",false,selectOption);
Weighted.setBounds(7,82,13,13);
Weighted.setBackground(DARKER_YELLOW);
add(Weighted);
Weighted.addItemListener(this);

weightedLabel = new Label ("Weighted", Label.LEFT);
weightedLabel.setBackground(DARKER_YELLOW);
weightedLabel.setForeground(LIGHT_BROWN);
weightedLabel.setBounds(23,83,60,12);
add(weightedLabel);

// CHOOSE A NEW SET OF LIBRARIES
setLibs = new Button("Set libraries");           // brings up window to choose
new set of libraries
setLibs.setBackground(DARKER_YELLOW);
setLibs.setForeground(DARK_BROWN);             // don't like how DARK_ORANGE
looks
setLibs.setBounds(370,10,90,20);
add(setLibs);
setLibs.addActionListener(new ActionListener() { public void
actionPerformed(ActionEvent ae)
{
    libsDone.setLabel("DONE"); // change
    frLibraries.setVisible(true);
}
});

back from "Wait..."

// DISPLAY CONTIGS IN THIS LIBRARY
// label for whichContigsFromLib choice list
libraryTitle = new Label("Show contigs in library:", Label.LEFT);
libraryTitle.setForeground(DARK_BROWN);
libraryTitle.setBounds(500,5,120,11);
add(libraryTitle);

// LOCATE AND DISPLAY INFORMATION ON CONTIG
// label for whereIsContig textfield
contigTitle = new Label("Enter contig #(s)", Label.LEFT);
contigTitle.setBackground(DARKER_YELLOW);
contigTitle.setForeground(DARK_ORANGE);
contigTitle.setBounds(642,5,125,11);
add(contigTitle);

whereAreContigs = new TextField(12);             // choose contig(s)
whereAreContigs.setBounds(637,18,125,28);
add(whereAreContigs);
// don't use TextListener: causes recalc as *each* character is entered!
whereAreContigs.addActionListener(new ActionListener() { public void
actionPerformed(ActionEvent e)
{

```

```

doWhereAreContigsProcess();
repaint();
    }
});

// upper text area to display component libraries & ESTs
whatsInContig = new TextArea("",11,47,1); // display component
libraries/ESTs: 11 rows, 47 chars, vertical only
whatsInContig.setBounds(635,50,280,194);
whatsInContig.setBackground(LIGHTER_YELLOW);
whatsInContig.setForeground(DARK_ORANGE); // color of text in text area
add(whatsInContig);

// DISPLAY NEW WINDOW WITH INFORMATION ON BEST BLAST HIT FOR CONTIG
bestHit = new Button ("Get Best Hit(s)");
bestHit.setBackground(getBackground());
bestHit.setForeground(DARK_ORANGE);
bestHit.setBounds(775,25,120,20);
add(bestHit);
bestHit.addActionListener(new ActionListener() { public void
actionPerformed(ActionEvent e)
{
    st = new StringTokenizer(whereAreContigs.getText()); // parse using space,
tab, \n or \r
    numInContigList = st.countTokens(); // count # contigs requested
    if ( numInContigList > 1 )
    {
        showBestHits();
    }
    else // one contig or nothing entered
    {
        if ( numInContigList == 1 )
        {
            contigNumber = st.nextToken(); // replaces
whereAreContigs.getText().trim() (leading/trailing spaces)
            showBestHit();
        }
        else // nothing entered
        {
            showStatus("You must specify a contig to get the best hit for a
contig");
        }
    }
    // END bestHit ActionListener
}
});

// SORT LIBRARIES
// label for libraryOrder choice list
sortTitle = new Label("Select category to sort by", Label.LEFT);
sortTitle.setBackground(DARKER_YELLOW);
sortTitle.setForeground(MEDIUM_GREEN);
sortTitle.setBounds(737,264,164,11);
add(sortTitle);

```

```

libraryOrder = new Choice();
libraryOrder.addItem(" tissue");
libraryOrder.addItem(" species");
libraryOrder.addItem(" germplasm");
libraryOrder.addItem(" dev_stage");
libraryOrder.addItem(" condition");
libraryOrder.addItem(" CUSTOM");
libraryOrder.setBounds(635,260,100,20);
libraryOrder.setBackground(LIGHTER_YELLOW); // color of drop-down
default=white (which is OK, too)
libraryOrder.setForeground(MEDIUM_GREEN); // color of text
add(libraryOrder);
libraryOrder.addItemListener(new ItemListener() { public void
itemStateChanged(ItemEvent ie)
{
    sortLibraries();
}
});

// middle text area to display library sort criteria
// e.g., Secale cereale: SC010XXX,SC013XXX,SC024E1X
// Triticum aestivum: TA001E1S, etc."
librarySortData = new TextArea(); // default is 0, both
scrollbars
librarySortData.setBounds(635,285,280,185);
librarySortData.setBackground(LIGHTER_YELLOW); // LIGHT_OLIVE too dark
librarySortData.setForeground(MEDIUM_GREEN); // color of text in text
area
add(librarySortData);

// LIST NAMES OF CONTIGS IN CLUSTER
contigList = new TextArea("",4,47,3); // 4 rows, 47 chars, 3=no scrollbars
// 0 or 4 is both, 1 or 2 is vertical only
contigList.setBounds(630,480,290,45);
contigList.setBackground(LIGHTER_YELLOW); // LIGHT_OLIVE too dark
add(contigList);

// end all GUI

// bring up window to ask user to choose libraries
// needs to be at end of init(); otherwise, frLibraries non-responsive until
init() done:
frLibraries.setVisible(true);

//err10="finished init(): "+System.currentTimeMillis();
} // end init()
//*****
*
// removed public void run() when I deleted thread
//*****
*
public void stop()
//*****
{

```

```

    // just in case frames not closed normally:
    frLibraries.setVisible(false); // close 'Choose libraries' window
    frHelp.setVisible(false);      // close help window
    frCustom1.setVisible(false);   // close custom sort windows
    frCustom2.setVisible(false);
}
//*****

    public void paint (Graphics g)
//*****

    {
        try
        {
            whatsInContig.setText(""); // to prevent duplicate output to textarea

            // CIRCLE
            g.setColor(DARK_GREEN);
            g.drawOval (START_X, START_Y, CIRCLE_RADIUS_INT*2, CIRCLE_RADIUS_INT*2); //
perimeter of circle
            g.setColor(Color.white);
            g.fillOval (START_X+1, START_Y+1, CIRCLE_RADIUS_INT*2-2, CIRCLE_RADIUS_INT*2-
2); // fill in circle

            // BOX AROUND RADIO BUTTONS
            g.setColor(DARKER_YELLOW);
            g.fill3DRect (3, 34, 88, 68, true); // 3D box around radio buttons

            // BOXES & LINES AROUND DATA AREAS
            g.setColor(DARKER_YELLOW);
            g.fill3DRect (628, 4, 293, 245, true); // raised rectangle around top set of
controls
            g.setColor(DARK_ORANGE);
            g.draw3DRect (627, 3, 295, 247, false); // shadow effect around top set of
controls

            g.setColor(DARKER_YELLOW); // lower set of controls
            g.fill3DRect (628, 256, 293, 218, true);
            g.setColor(MEDIUM_GREEN);
            g.draw3DRect (627, 255, 295, 220, false); // shadow effect around bottom
set of controls

            // START CODE for IF NOT the first paint()
            if ( !initialLoad && // OK to do rest of GUI if not initial
applet invocation &
                !frLibraries.isShowing() && // pop-up windows aren't showing
                !frCustom1.isShowing() &&
                !frCustom2.isShowing() )
            {
                // CONTIG DOTS INSIDE CIRCLE
                g.setColor(Color.green); // WHITE_YELLOW too light, MEDIUM_GREEN TOO
DARK

                if (showEqual) // "Equal" radio button selected (default)
                {

```

```

        for (int i = 0; i < numberOfContigs; i++)
        {
            g.fillOval(xCoordEContig[i]-2,yCoordEContig[i]-2,5,5 );
        }
    }
    else if (showProportional) // "Proportional" radio button selected
    {
        for (int i = 0; i < numberOfContigs; i++)
        {
            g.fillOval(xCoordPContig[i]-2, yCoordPContig[i]-2,5,5 );
        }
    }
    else // "Weighted" radio button selected
    {
        for (int i = 0; i < numberOfContigs; i++)
        {
            g.fillOval(xCoordWContig[i]-2,yCoordWContig[i]-2,5,5 );
        }
    } // end if showEqual

    // CENTER DOT
    g.setColor(Color.red);
    g.drawOval(xCircle-1,yCircle-1,4,4); // center dot

    // LIBRARY DOTS
    drawLibraries(g); // draw points (representing libraries) on
circle

    // if mouse pointer CLICKED, draw LINES from contig to contributing
libraries
    // if (clickMouseX > 0)
    // {
    //     if ( !clickedContig.equals("") ) // there is actually a contig at
this location
    //     {
    //         g.setColor(Color.black);
    //         g.fillOval(clickMouseX-3,clickMouseY-3,7,7);

    //         if (!lookingForContig) // nothing in contig textField
    //         {
    //             drawLibraryLines(g,clickMouseX,clickMouseY,clickedContig); // draw
lines from clicked contig
    //         }
    //         else // user has also requested a contig in textbox
    //         {
    //             // show previous (clicked) contig in relationship to
newly requested contig

    g.drawString(""+clickedContig+clickedMode,clickMouseX,clickMouseY); // just
label the point; no lines
    //         getClickedContigData(); // get lib/ESTs for clicked
contig from which no lines'll be drawn
    //     }
    // }

```

```

        whatsInContig.append(contigDataString);    // write lib/ESTs to
textarea
    }
else if (clickMouseX > 0) // user clicked, but no such contig at this x,y
location
//
    else // no such contig at this x,y location
    {
        g.setColor(Color.red);
        g.drawString("Click point does not match a contig",630,549);
    } // end if clickedContig not null
//
    } // end if clickMouseX > 0

// display library sort info in middle text area
librarySortData.setText(""); // clear prior sort output
librarySortData.append(librarySortString);
librarySortData.append("\n");

// if user has dragged and released mouse, draw a RECTANGLE around the
cluster
if (drawRectAroundContigs)
{
    g.setColor(Color.black);
    g.drawRect(pressMouseX,pressMouseY,(releaseMouseX-pressMouseX),
                (releaseMouseY-pressMouseY));
}

// list selected contigs in lower text area
contigList.setText(""); // clear prior output
if (getListOfContigs)
{
    contigList.append(contigListString);
}
else // nothing selected or libraries just resorted
{
    contigList.append("To select and list several contigs, point mouse
at\n");
    contigList.append("upper left corner of cluster, press, drag,
then\n");
    contigList.append("release mouse at lower right corner of cluster.");
}

//SUMMARY info in upper left corner
g.setColor(DARK_BROWN);

g.drawString(numberOfESTs+ " ESTs from",10,10);
g.drawString(numberOfLibraries+" librar"+(numberOfLibraries > 1 ? "ies"
: "y")+ " assembled into",10,20);
g.drawString(numberOfContigs+" contigs",10,30);

// "HIGHLIGHT CONTIG" section should precede "Display all contigs" block
// so that libraryRequested highlighted

```

```

// HIGHLIGHT CONTIG THAT USER IS LOOKING FOR and draw lines out to
libraries
if (lookingForContig) // at least one contig entered in
"whereAreContigs" TextField
{
    int xCoord;
    int yCoord;

    g.setColor(Color.red);

    if ( (showEqual) && (iContig >= 0) ) // EQUAL (default) & contig
found
    {
        xCoord=xCoordeContig[iContig];
        yCoord=yCoordeContig[iContig];

        g.fillOval(xCoord-2,yCoord-2,5,5);
        drawLibraryLines(g,xCoord,yCoord,contigToShow);
        whatsInContig.append(contigDataString); // write lib/ESTs to
upper text area
    }
    else if ( (showProportional) && (iContig >= 0) ) //
PROPORTIONAL & contig found
    {
        xCoord=xCoordPContig[iContig];
        yCoord=yCoordPContig[iContig];

        g.fillOval(xCoord-2,yCoord-2,5,5); // -2 so that lines and
dot connect
        drawLibraryLines(g,xCoord,yCoord,contigToShow); // connect contigs
to libraries
        whatsInContig.append(contigDataString); // write lib/ESTs for
requested contig to upper text area
    }
    else if ( (showWeighted) && (iContig >= 0) ) // WEIGHTED & contig
found
    {
        xCoord=xCoordWContig[iContig];
        yCoord=yCoordWContig[iContig];

        g.fillOval(xCoord-2,yCoord-2,5,5);
        drawLibraryLines(g,xCoord,yCoord,contigToShow);
        whatsInContig.append(contigDataString); // write lib/ESTs to
upper text area
    }
    else // contig not found (iContig = -
1)
    {
        g.drawString(contigToShow+" not found",770,14);
    }
} // end if lookingForContig

// HIGHLIGHT CONTIGS IN LIST (NO LINES TO LIBRARIES )
if (lookingForContigs) // list entered in "whereAreContigs" text field
{

```

```

g.setColor(Color.black);

if (showEqual)                                // default
{
    for (int i=1; i<numInContigList; i++)      // only need to do this
for 2nd thru last contig
    {
        if (iContigs[i] >= 0)
        {
            g.drawOval(xCoordEContig[iContigs[i]]-2,
                        yCoordEContig[iContigs[i]]-2,5,5);

g.drawString(""+contigNames[iContigs[i]],xCoordEContig[iContigs[i]],
                                yCoordEContig[iContigs[i]]);
        } // end if
    } // end for
}
else if (showProportional)
{
    for (int i=1; i<numInContigList; i++)      // only need to do this
for 2nd thru last contig
    {
        if (iContigs[i] >= 0)
        {
            g.drawOval(xCoordPContig[iContigs[i]]-2,
                        yCoordPContig[iContigs[i]]-2,5,5);

g.drawString(""+contigNames[iContigs[i]],xCoordPContig[iContigs[i]],
                                yCoordPContig[iContigs[i]]);
        } // end if
    } // end for
}
else                                           // if (showWeighted)
{
    for (int i=1; i<numInContigList; i++)      // only need to do this
for 2nd thru last contig
    {
        if (iContigs[i] >= 0)
        {
            g.drawOval(xCoordWContig[iContigs[i]]-2,
                        yCoordWContig[iContigs[i]]-2,5,5);

g.drawString(""+contigNames[iContigs[i]],xCoordWContig[iContigs[i]],
                                yCoordWContig[iContigs[i]]);
        } // end if
    } // end for
} // end if showProp/Equal/Wtd

addToContigDataString();
whatsInContig.append(contigDataString);

if (!allContigsFound)
{
    g.drawString("contig(s) not found", 770,23);
}

} // end if lookingForContigs

```

```

// DISPLAY ALL CONTIGS IN SELECTED LIBRARY #1
//*****
if (showContigsInLib1)          // something chosen in library pull-down
{
    int numContigs = contigsInLibrary1.length;

    // paint contigs from requested library
    if (showEqual)                // show EQUAL (default) library
representation
    {
        g.setColor(DARK_RED);
        g.fillOval(610,26,4,4);

        for (int i=0; i<numContigs; i++)          // go thru library's
contigs
        {
            for (int j=0; j<numberOfContigs; j++) // go thru all contigs
available
            {
                if ( contigsInLibrary1[i].equals(contigNames[j]) )
                {
                    g.fillOval(xCoordEContig[j]-2,yCoordEContig[j]-2,4,4);
                    break; // break out of inner "for" loop
                }
            } // end inner "for j" loop
        } // end outer "for i" loop
    }
    else if (showProportional)                // proportional library
representation
    {
        g.setColor(LIGHT_OLIVE);
        g.fillOval(610,26,4,4);                // legend next to choice list

        for (int i=0; i<numContigs; i++)          // go thru library's
contigs
        {
            for (int j=0; j<numberOfContigs; j++) // go thru all contigs
available
            {
                if ( contigsInLibrary1[i].equals(contigNames[j]) )
                {
                    g.fillOval(xCoordPContig[j]-2,yCoordPContig[j]-2,4,4);
                    break; // break out of inner "for" loop
                }
            } // end inner "for j" loop
        } // end outer "for i" loop
    }
    else
representation                                // show WEIGHTED library
    {
        g.setColor(LIGHT_BROWN);
        g.fillOval(610,26,4,4);

        for (int i=0; i<numContigs; i++)          // go thru library's
contigs
        {

```

```

        for (int j=0; j<numberOfContigs; j++)    // go thru all contigs
available
    {
        if ( contigsInLibrary1[i].equals(contigNames[j]) )
        {
            g.fillOval(xCoordWContig[j]-2,yCoordWContig[j]-2,4,4);
            break; // break out of inner "for" loop
        }
    } // end inner "for j" loop
    } // end outer "for i" loop
    } // if showProportional

    // label requested library    // DARKER_YELLOW, ORANGE too light

g.drawString(""+sortedLibraryNames[indexRequestedLib1].trim(),labelX[indexReques
tedLib1],

                                                    labelY[indexRequestedLib1]);

    } // end if showContigsInLib1
    //*****
    // DISPLAY ALL CONTIGS IN SELECTED LIBRARY #2
    //*****
    if (showContigsInLib2)    // something chosen in library pull-down
    {
        int numContigs = contigsInLibrary2.length;

        // paint contigs from requested library
        if (showEqual)    // show EQUAL (default) library
representation
        {
            g.setColor(DARK_RED);
            g.drawOval(610,45,5,5);

            for (int i=0; i<numContigs; i++)    // go thru library's
contigs
            {
                for (int j=0; j<numberOfContigs; j++)    // go thru all contigs
available
                {
                    if ( contigsInLibrary2[i].equals(contigNames[j]) )
                    {
                        g.drawOval(xCoordEContig[j]-2,yCoordEContig[j]-2,5,5);
                        break; // break out of inner "for" loop
                    }
                } // end inner "for j" loop
            } // end outer "for i" loop
        }
        else if (showProportional)    // proportional library
representation
        {
            g.setColor(LIGHT_OLIVE);
            g.drawOval(610,45,5,5);

            for (int i=0; i<numContigs; i++)    // go thru library's
contigs
            {

```

```

        for (int j=0; j<numberOfContigs; j++)    // go thru all contigs
available
    {
        if ( contigsInLibrary2[i].equals(contigNames[j]) )
        {
            g.drawOval(xCoordPContig[j]-2,yCoordPContig[j]-2,5,5);
            break; // break out of inner "for" loop
        }
    } // end inner "for j" loop
    } // end outer "for i" loop
}
else // show WEIGHTED library
representation
{
    g.setColor(LIGHT_BROWN);
    g.drawOval(610,45,5,5);

    for (int i=0; i<numContigs; i++)            // go thru library's
contigs
    {
        for (int j=0; j<numberOfContigs; j++)    // go thru all contigs
available
        {
            if ( contigsInLibrary2[i].equals(contigNames[j]) )
            {
                g.drawOval(xCoordWContig[j]-2,yCoordWContig[j]-2,5,5);
                break; // break out of inner "for" loop
            }
        } // end inner "for j" loop
    } // end outer "for i" loop
    } // if showProportional

    // label requested library    // DARKER_YELLOW, ORANGE too light

g.drawString(""+sortedLibraryNames[indexRequestedLib2].trim(),labelX[indexReques
tedLib2],
                                                    labelY[indexRequestedLib2]);
    } // end if showContigsInLib2
//*****
// DISPLAY ALL CONTIGS IN SELECTED LIBRARY #3
//*****
if (showContigsInLib3)    // something chosen in library pull-down
{
    int numContigs = contigsInLibrary3.length;

    // paint contigs from requested library
    if (showEqual)    // show EQUAL (default) library
representation
    {
        g.setColor(DARK_RED);
        g.drawRect(610,64,5,5);

        for (int i=0; i<numContigs; i++)            // go thru library's
contigs
        {
            for (int j=0; j<numberOfContigs; j++)    // go thru all contigs
available

```

```

        {
            if ( contigsInLibrary3[i].equals(contigNames[j]) )
            {
                g.drawRect(xCoordEContig[j]-2,yCoordEContig[j]-2,5,5);
                break; // break out of inner "for" loop
            }
        } // end inner "for j" loop
    } // end outer "for i" loop
}
else if (showProportional) // proportional library
representation
{
    g.setColor(LIGHT_OLIVE);
    g.drawRect(610,64,5,5);

    for (int i=0; i<numContigs; i++) // go thru library's
contigs
    {
        for (int j=0; j<numberOfContigs; j++) // go thru all contigs
available
        {
            if ( contigsInLibrary3[i].equals(contigNames[j]) )
            {
                g.drawRect(xCoordPContig[j]-2,yCoordPContig[j]-2,5,5);
                break; // break out of inner "for" loop
            }
        } // end inner "for j" loop
    } // end outer "for i" loop
}
else // show WEIGHTED library
representation
{
    g.setColor(LIGHT_BROWN);
    g.drawRect(610,64,5,5);

    for (int i=0; i<numContigs; i++) // go thru library's
contigs
    {
        for (int j=0; j<numberOfContigs; j++) // go thru all contigs
available
        {
            if ( contigsInLibrary3[i].equals(contigNames[j]) )
            {
                g.drawRect(xCoordWContig[j]-2,yCoordWContig[j]-2,5,5);
                break; // break out of inner "for" loop
            }
        } // end inner "for j" loop
    } // end outer "for i" loop
} // if showProportional

    // label requested library // DARKER_YELLOW, ORANGE too light

g.drawString(""+sortedLibraryNames[indexRequestedLib3].trim(),labelX[indexReques
tedLib3],
labelY[indexRequestedLib3]);

} // end if showContigsInLib3

```

```

//*****

// to show user that processing is complete:
g.setColor(DARK_BROWN);
g.drawString("DONE", 630,560);
}
// END CODE for IF NOT the first paint()
else // initialLoad
{
    contigList.append("\nAwaiting user selections, sorting\n");
    contigList.append("of libraries, and calculation of\n");
    contigList.append("library contributions to contigs...");

// for debug: contigList.append(""+err1+"\n");

    } // end if (initialLoad)
}
catch (Exception e)
{
    err1=err1+" paint() error: "+e;
}

//debug start

//use for try/catch
//use in final applet:
g.drawString("Error: "+err1,10,565);

//use for timing      g.drawString("err2 "+err2,620,540);
//use for timing      g.drawString("err3 "+err3,620,550);
//use for timing      g.drawString("err9 "+err9,620,560);
//err10="finished paint: "+System.currentTimeMillis();
//use for timing      g.drawString("err10 "+err10,620,570);

//debug end

} // end paint()
//*****

// public void update(Graphics g)
// {
// } // use default update (default background, then call paint(g))
//*****

void getKeySQLData() // "throws Exception" not needed because DataServer
methods already throw
// exception and I've "caught" it below
//*****

{
    try
    {
        server=new DataClient();

        // LIBRARIES
        origLibraryNames = server.getDistinctLibraryNames(); // array of strings
(library names)

```

```

origNumberOfLibraries = origLibraryNames.length;

// CONTIGS
// consider removing; used only to initialize contigHolder vector.
origNumberOfContigs = server.getDistinctContigNumber();

// get LIB_INFO for sorting later      // get library names from sort
criteria table
    LibInfo = server.getLibInfo();
    numberOfSortableLibs = LibInfo.length;

    Species =      server.getSpecies();      // get sort criteria for
all libraries
    Germplasm =    server.getGermplasm();
    Tissue =       server.getTissue();
    Stage =        server.getStage();
    Condition =    server.getCondition();

    // get total number of ESTs for each library
    origESTCountByLib = server.getESTCountOfLibs(); // get total #ESTs by
library
                                                    // replaces
server.getTotalESTNumber();

    } // end try
    catch (Exception e)
    {
        err1=err1+"getKeySQLData error: "+e;      // same as:
err1="" + e.getMessage();
// use for appletviewer debug:
e.printStackTrace();
// NB appletviewer does not support URL class (so comment out getBestHit() if
doing appletviewer debug)
    } // end catch
    } // end getKeySQLData()
//*****
void makeSortedLibraries() // populate libraryNamesBy___ arrays
//*****

{
    try
    {
        // lib_info tables (w/sort feature data) have data on libraries not
        represented in contigs;
        // therefore, need to extract feature data for those libraries actually
        represented in contigs:

        for (int i=0; i< numberOfLibraries; i++)
        {
            libraryNamesBySpecies[i]="";
            libraryNamesByGermplasm[i]="";
            libraryNamesByTissue[i]="";
            libraryNamesByStage[i]="";
            libraryNamesByCondition[i]="";

            for (int j=0; j< numberOfSortableLibs; j++) // includes non-contigged
libs

```

```

    {
        if ( libraryNames[i].trim().equals(LibInfo[j].trim()) )
            // need trim() because of \r in est table
        {
            speciesNames[i]=      Species[j];          // speciesNames = subset of
Species
            germplasmNames[i] = Germplasm[j];
            tissueNames[i] =      Tissue[j];
            stageNames[i] =       Stage[j];
            conditionNames[i] = Condition[j].trim();
            break; // break out of inner "for" loop, since lib found
        } // if
    } // for j
} // for i

// put sort feature arrays in alphanumeric order
// make a copy of germplasmNames[]
System.arraycopy(germplasmNames,0,sortedGermplasm,0,numberOfLibraries);
Arrays.sort(sortedGermplasm);

// go through sortedGermplasm[], populate libraryNamesByGermplasm[]
(libraries in germplasm order)
for (int i =0; i< numberOfLibraries; i++) // i - unsorted species/lib
{
    for (int j=0; j< numberOfLibraries; j++) // j - sorted species/lib
    {
        if ( sortedGermplasm[j].equals(germplasmNames[i]) &&
            libraryNamesByGermplasm[j].equals("") ) // if > 1 of same
germplasm
        {
            libraryNamesByGermplasm[j] = libraryNames[i];
            estCountByGermplasm[j] = estCountByLib[i];
            break; // go back to outer "for" loop
        }
    } // end for j-sorted
} // for i-unsorted

// make a copy of speciesNames[]
System.arraycopy(speciesNames,0,sortedSpecies,0,numberOfLibraries);
Arrays.sort(sortedSpecies);

// go through sortedSpecies[], populate libraryNamesBySpecies[]
for (int i =0; i< numberOfLibraries; i++) // i - unsorted
species/lib
{
    for (int j=0; j< numberOfLibraries; j++) // j - sorted species/lib
    {
        if ( sortedSpecies[j].equals(speciesNames[i]) &&
            libraryNamesBySpecies[j].equals("") ) // if > 1 of same species
        {
            libraryNamesBySpecies[j] = libraryNames[i];
            estCountBySpecies[j] = estCountByLib[i];
            break; // go back to outer "for" loop
        }
    } // end for j-sorted
} // for i-unsorted

```

```

// make a copy of tissueNames[]
System.arraycopy(tissueNames,0,sortedTissue,0,numberOfLibraries);
Arrays.sort(sortedTissue);

// go through sortedTissue[], populate libraryNamesByTissue[]
for (int i =0; i< numberOfLibraries; i++)          // i - unsorted tissue/lib
{
    for (int j=0; j< numberOfLibraries; j++)        // j - sorted tissue/lib
    {
        if ( sortedTissue[j].equals(tissueNames[i]) &&
            libraryNamesByTissue[j].equals("") )    // if > 1 of same tissue
        {
            libraryNamesByTissue[j] = libraryNames[i];
            estCountByTissue[j] = estCountByLib[i];
            break; // go back to outer "for" loop
        }
    } // end for j-sorted
} // for i-unsorted

// make a copy of stageNames[]
System.arraycopy(stageNames,0,sortedStage,0,numberOfLibraries);
Arrays.sort(sortedStage);

// go through sortedStage[], populate libraryNamesByStage[]
for (int i =0; i< numberOfLibraries; i++)          // i - unsorted stage/lib
{
    for (int j=0; j< numberOfLibraries; j++)        // j - sorted stage/lib
    {
        if ( sortedStage[j].equals(stageNames[i]) &&
            libraryNamesByStage[j].equals("") )    // if > 1 of same stage
        {
            libraryNamesByStage[j] = libraryNames[i];
            estCountByStage[j] = estCountByLib[i];
            break; // go back to outer "for" loop
        }
    } // end for j-sorted
} // for i-unsorted

// make a copy of conditionNames[]
System.arraycopy(conditionNames,0,sortedCondition,0,numberOfLibraries);
Arrays.sort(sortedCondition);

// go through sortedCondition[], populate libraryNamesByCondition[]
for (int i =0; i< numberOfLibraries; i++)          // i - unsorted
condition/lib
{
    for (int j=0; j< numberOfLibraries; j++)        // j - sorted
condition/lib
    {
        if ( sortedCondition[j].equals(conditionNames[i]) &&
            libraryNamesByCondition[j].equals("") ) // if > 1 of same
condition
        {

```

```

        libraryNamesByCondition[j] = libraryNames[i];
        estCountByCondition[j] = estCountByLib[i];
        break; // go back to outer "for" loop
    }
} // end for j-sorted
} // for i-unsorted
}
catch (Exception e)
{
    err1+="makeSortedLib: "+e;
}
} // end makeSortedLibraries()
//*****
void calculateSortedLibContribution() // use if user wants libraries sorted
//*****

{
    // 'synchronized' unnecessary since only one call to this function at a time
    //
    // didn't do separate thread for following because don't want GUI to be
    // responsive
    // to user changes during sort.
    //
    // (contig points have to be recalculated every time libraries sorted,
    // because indiv "pContrib/eContrib" data not captured)
    //
    // calculate pull of libraries on contig (pContribFromLibrary,
    eContribFromLibrary)
    // then populate xCoordPContig, xCoordEContig vectors with x,y coordinates for
    contigs

    String lib;
    int numESTs;
    int libIndex = -1;
    contigCount = 0;

    try
    {
        for (int i=0; i < numberOfContigs; i++)
        {
            // err2="start calculateSortedLibContrib "+System.currentTimeMillis();
            // next line takes the most time: 7 milliseconds x 8000 = 56 seconds
            estObjs = server.getESTNumberOfLibrary(contigNames[i]);
            // err3="after sorted estObjs assignment "+System.currentTimeMillis();

            totalLibsInContig = estObjs.length; // # libraries in each
            contig, used to search current library set

            // cycle through array of EstsOfLibrary (library + #ESTs) objects for
            this contig:
            for (int j=0; j < totalLibsInContig; j++) // same as estObjs.length
            {
                lib = estObjs[j].getLibName(); // name of library (may or may
                not be in current selection)
                numESTs = estObjs[j].getNumOfEsts(); // number of ESTs from
                that library
            }
        }
    }
    catch (Exception e)
    {
        err1+="calculateSortedLibContrib: "+e;
    }
}

```

```

        for (int k=0; k < numberOfLibraries; k++)    // cycle through current
selection of libraries
    {
        if ( lib.equals(sortedLibraryNames[k]) ) // library in contig
matches library in master set
        {
            libIndex = k;                        // capture array index

            // find library in this contig with the highest number of ESTs -
for "wSizeOfMovement"
            if (j>0)
            {
                if (estCountBySortedLib[libIndex] >
estCountBySortedLib[indexMostESTs])
                {
                    indexMostESTs=libIndex;
                }
            }
            else    // j==0: assume first library has most ESTs
            {
                indexMostESTs=libIndex;
            }

            eLibsPresent++;    // # libraries in contig also in current lib
selection, for "eSizeOfMovement"

            pNumESTsPresent += numESTs;    // accumulate # ESTs in contig;
used for "pSizeOfMovement"

            // increment only if lib in contig also in
curr lib selection
            switch (libIndex)
            {
                case 0:
                    pContribFromLibrary[0] = numESTs; // PROPORTIONAL
                    eContribFromLibrary[0] = 1;        // EQUAL
                    break;
                case 1:
                    pContribFromLibrary[1] = numESTs;
                    eContribFromLibrary[1] = 1;
                    break;
                case 2:
                    pContribFromLibrary[2] = numESTs;
                    eContribFromLibrary[2] = 1;
                    break;
                case 3:
                    pContribFromLibrary[3] = numESTs;
                    eContribFromLibrary[3] = 1;
                    break;
                case 4:
                    pContribFromLibrary[4] = numESTs;
                    eContribFromLibrary[4] = 1;
                    break;
                case 5:
                    pContribFromLibrary[5] = numESTs;
                    eContribFromLibrary[5] = 1;
                    break;
                case 6:

```

```

    pContribFromLibrary[6] = numESTs;
    eContribFromLibrary[6] = 1;
    break;
case 7:
    pContribFromLibrary[7] = numESTs;
    eContribFromLibrary[7] = 1;
    break;
case 8:
    pContribFromLibrary[8] = numESTs;
    eContribFromLibrary[8] = 1;
    break;
case 9:
    pContribFromLibrary[9] = numESTs;
    eContribFromLibrary[9] = 1;
    break;
case 10:
    pContribFromLibrary[10] = numESTs;
    eContribFromLibrary[10] = 1;
    break;
case 11:
pContribFromLibrary[11] = numESTs;
    eContribFromLibrary[11] = 1;
    break;
case 12:
    pContribFromLibrary[12] = numESTs;
    eContribFromLibrary[12] = 1;
    break;
case 13:
    pContribFromLibrary[13] = numESTs;
    eContribFromLibrary[13] = 1;
    break;
case 14:
    pContribFromLibrary[14] = numESTs;
    eContribFromLibrary[14] = 1;
    break;
case 15:
    pContribFromLibrary[15] = numESTs;
    eContribFromLibrary[15] = 1;
    break;
case 16:
    pContribFromLibrary[16] = numESTs;
    eContribFromLibrary[16] = 1;
    break;
case 17:
    pContribFromLibrary[17] = numESTs;
    eContribFromLibrary[17] = 1;
    break;
case 18:
    pContribFromLibrary[18] = numESTs;
    eContribFromLibrary[18] = 1;
    break;
case 19:
    pContribFromLibrary[19] = numESTs;
    eContribFromLibrary[19] = 1;
    break;
case 20:
    pContribFromLibrary[20] = numESTs;

```

```

        eContribFromLibrary[20] = 1;
        break;
    case 21:
        pContribFromLibrary[21] = numESTs;
        eContribFromLibrary[21] = 1;
        break;
    case 22:
        pContribFromLibrary[22] = numESTs;
        eContribFromLibrary[22] = 1;
        break;
    case 23:
        pContribFromLibrary[23] = numESTs;
        eContribFromLibrary[23] = 1;
        break;
    case 24:
        pContribFromLibrary[24] = numESTs;
        eContribFromLibrary[24] = 1;
        break;
    case 25:
        pContribFromLibrary[25] = numESTs;
        eContribFromLibrary[25] = 1;
        break;
    case 26:
        pContribFromLibrary[26] = numESTs;
        eContribFromLibrary[26] = 1;
        break;
    case 27:
        pContribFromLibrary[27] = numESTs;
        eContribFromLibrary[27] = 1;
        break;
    /** REMOVE COMMENT LATER, when there are more than 28 libraries
    case 28:
        pContribFromLibrary[28] = numESTs;
        eContribFromLibrary[28] = 1;
        break;
    case 29:
        pContribFromLibrary[29] = numESTs;
        eContribFromLibrary[29] = 1;
        break;
    case 30:
        pContribFromLibrary[30] = numESTs;
        eContribFromLibrary[30] = 1;
        break;
    case 31:
        pContribFromLibrary[31] = numESTs;
        eContribFromLibrary[31] = 1;
        break;
    case 32:
        pContribFromLibrary[32] = numESTs;
        eContribFromLibrary[32] = 1;
        break;
    case 33:
        pContribFromLibrary[33] = numESTs;
        eContribFromLibrary[33] = 1;
        break;

    */
    default:

```

```

        } // end switch
    } // end if lib equals sortedLibraryName
} // end for numberOfLibraries (current library selection)
} // end for each library-#EST array element

calculateXY(); // calculate and save x,y for each contig

reinitializeContribs();

} // end for each contig
}
catch (Exception e)
{
    err1+="calcSortedLibContrib error: "+e;
}
} // end calculateSortedLibContribution()
//*****
void calculateXY()
//*****

{
    // calculate x,y for contig point, reflecting sum of library contributions to
    contig
    try
    {
        pSizeOfMovement = CIRCLE_RADIUS / pNumESTsPresent; // unit of 'pull' depends
on #ESTs from lib present
        eSizeOfMovement = CIRCLE_RADIUS / eLibsPresent; // unit of 'pull' = 1 for
each lib present

        // START calculations for weighted contigs

        for (int i=0; i<numberOfLibraries; i++)
        {
            // libraries with more total ESTs "pull" less
            wContribFromLibrary[i] = (int)(
(pContribFromLibrary[i]*estCountBySortedLib[indexMostESTs]) /
estCountBySortedLib[i] );

            // total up contributions
            wContigSize += wContribFromLibrary[i];
        }

        wSizeOfMovement = CIRCLE_RADIUS / wContigSize; // unit of 'pull' dep on
#ESTs & total ESTs
        // END calculations for weighted contigs

        // need to break up calculateXY() because size can't exceed 65,000 bytes
        if ( ( 1 <= numberOfLibraries) &&
            ( 15 >= numberOfLibraries) )
        {
            calcXY_1to15();
        }
        else if ( ( 16 <= numberOfLibraries) &&
            ( 21 >= numberOfLibraries) )
        {
            calcXY_16to21();
        }
    }
}

```

```

    }
    else if ( ( 22 <= numberOfLibraries) &&
              ( 26 >= numberOfLibraries) )
    {
        calcXY_22to26();
    }
    else if ( ( 27 <= numberOfLibraries) &&
              ( 30 >= numberOfLibraries) )
    {
        calcXY_27to30();
    }
}
// REMOVE COMMENTS FOR 29-34 LATER, when there are more than 28 libraries
/**
    else if ( ( 31 <= numberOfLibraries) &&
              ( 33 >= numberOfLibraries) )
    {
        calcXY_31to33();
    }
    else if ( 34 == numberOfLibraries )
    {
        calcXY_34();
    }
*/

// populate arrays of x,y coordinates
xCoordPContig[contigCount]=xPContig;// x-coordinate for PROPORTIONAL contig
yCoordPContig[contigCount]=yPContig;// y-coordinate for PROPORTIONAL contig
xCoordEContig[contigCount]=xEContig;// x-coordinate for EQUAL contig
yCoordEContig[contigCount]=yEContig;// y-coordinate for EQUAL contig
xCoordWContig[contigCount]=xWContig;// x-coordinate for WEIGHTED contig
yCoordWContig[contigCount++]=yWContig;// y-coordinate for WEIGHTED contig
}
catch (Exception e)
{
    err1+="calculateXY() error: "+e;
}

} // end calculateXY()
//*****
void calcXY_1to15()
//*****

{
    if ( 1 == numberOfLibraries)
    {
        xPContig=xCircle+3;           // slightly offset from "library" dot
        yPContig=yCircle-(int) (pContribFromLibrary[0]*pSizeOfMovement)+3;
        xEContig=xCircle+3;           // slightly offset from "library" dot
        yEContig=yCircle-(int) (eContribFromLibrary[0]*eSizeOfMovement)+3;
        xWContig=xCircle+3;           // slightly offset from "library" dot
        yWContig=yCircle-(int) (wContribFromLibrary[0]*wSizeOfMovement)+3;

    } // end if numberOfLibraries is 1

    else if ( 2 == numberOfLibraries)
    {
        xPContig=xCircle;
        yPContig=yCircle

```

```

        -(int) (pContribFromLibrary[0]*pSizeOfMovement)

        +(int) (pContribFromLibrary[1]*pSizeOfMovement);
    xEContig=xCircle;
    yEContig=yCircle
        -(int) (eContribFromLibrary[0]*eSizeOfMovement)

        +(int) (eContribFromLibrary[1]*eSizeOfMovement);
    xWContig=xCircle;
    yWContig=yCircle
        -(int) (wContribFromLibrary[0]*wSizeOfMovement)

        +(int) (wContribFromLibrary[1]*wSizeOfMovement);

    } // end else if numberOfLibraries is 2

// START DOCUMENTATION FOR TRIGONOMETRIC CALCULATION
// Place an angle of 'a' degrees with vertex at the center of the circle
// and one side along the horizontal axis.
// The endpoint of the other side of the angle lies along the circle, and
// this point is 'x' horizontal distance and 'y' vertical distance from the
// circle's
// center. If a line is drawn directly from the center to the point (picture
// the
// hypotenuse of a right triangle) with the length 'z,' then  $\sin a = y/z$ ,  $\cos a$ 
//  $= x/z$ .
//
// To place a contig at the proper location, we need to get the values of 'x'
// and 'y.'
//  $x = (\cos a)*z$ 
//  $y = (\sin a)*z$ 
//
// First, we need to calculate the angle measurement 'a':
// Divide 360 by number of libraries to get the # degrees for each library
// angle.
// Draw lines out from center of circle to each library.
// Draw a horizontal line through the center of the circle.
// Calculate how far (#degrees in the angle 'a') each library is from this
// horizontal line.
//
// Next, we use 'a' to determine the value of 'x' and 'y':
// Java requires that the angle measurement be expressed in radians, not
// degrees.
// see
// http://java.sun.com/j2se/1.4.1/docs/api/java/lang/Math.html#cos\(double\)
// A radian is the length of the arc cut off by the angle.
// To convert from degrees to radians:
// Circumference of a circle of radius 1 is  $2*\pi$ , so 360 degrees =  $2\pi$  radians.
// 1 degree =  $(\pi/180)$  radians
// trig lesson from: http://aleph0.clarku.edu/~djoyce/java/trig/angle.html
//
// 30 degrees =  $30*\text{Math.PI}/180$  radians [pi = Math.PI in Java]
// If 'a' is 30 degrees, then
//  $\cos a = \text{Math.cos}(30*\text{Math.PI}/180)$  [cos = Math.cos in Java]
//  $x = (\cos a)*z$  translated to Java is  $x = \text{Math.cos}(30*\text{Math.PI}/180) * z$ 
// replacing 'z' with each library (i)'s pull:
    (pContribFromLibrary[i]*pSizeOfMovement)

```

```

//
// The same logic applies to 'y' and the sine of the angle.
//
// Use + (add) for 'x' if the library is right of center, - (subtract) if left
of center.
// Use + for 'y' if the library is below the center, - if above the center.
// END DOCUMENTATION FOR TRIGONOMETRIC CALCULATION

    else if ( 3 == numberOfLibraries)
    {
        xPContig=xCircle
+ (int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement));
        yPContig=yCircle
- (int) (pContribFromLibrary[0] * pSizeOfMovement)

+ (int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement));
        xEContig=xCircle

+ (int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement));
        yEContig=yCircle
- (int) (eContribFromLibrary[0] * eSizeOfMovement)

+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement));
        xWContig=xCircle

+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement));
        yWContig=yCircle
- (int) (wContribFromLibrary[0] * wSizeOfMovement)

+ (int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement));

    } // end else if numberOfLibraries is 3

    else if ( 4 == numberOfLibraries)
    {
        xPContig=xCircle
+ (int) (pContribFromLibrary[1] * pSizeOfMovement)
- (int) (pContribFromLibrary[3] * pSizeOfMovement);
        yPContig=yCircle
- (int) (pContribFromLibrary[0] * pSizeOfMovement)
+ (int) (pContribFromLibrary[2] * pSizeOfMovement);

```

```

    xEContig=xCircle
        +(int) (eContribFromLibrary[1]*eSizeOfMovement)
        -(int) (eContribFromLibrary[3]*eSizeOfMovement);
    yEContig=yCircle
        -(int) (eContribFromLibrary[0]*eSizeOfMovement)
        +(int) (eContribFromLibrary[2]*eSizeOfMovement);
    xWContig=xCircle
        +(int) (wContribFromLibrary[1]*wSizeOfMovement)
        -(int) (wContribFromLibrary[3]*wSizeOfMovement);
    yWContig=yCircle
        -(int) (wContribFromLibrary[0]*wSizeOfMovement)
        +(int) (wContribFromLibrary[2]*wSizeOfMovement);
} // end if numberOfLibraries is 4

// if numberOfLibraries is 5, 72 degrees per slice
else if ( 5 == numberOfLibraries)
{
    xPContig=xCircle

    +(int) (Math.cos(18*Math.PI/180)*(pContribFromLibrary[1]*pSizeOfMovement))
    +(int) (Math.cos(54*Math.PI/180)*(pContribFromLibrary[2]*pSizeOfMovement))
    -
    (int) (Math.cos(54*Math.PI/180)*(pContribFromLibrary[3]*pSizeOfMovement))
    -
    (int) (Math.cos(18*Math.PI/180)*(pContribFromLibrary[4]*pSizeOfMovement));
    yPContig=yCircle
        -(int) (pContribFromLibrary[0]*pSizeOfMovement)
        -
    (int) (Math.sin(18*Math.PI/180)*(pContribFromLibrary[1]*pSizeOfMovement))
    +(int) (Math.sin(54*Math.PI/180)*(pContribFromLibrary[2]*pSizeOfMovement))
    +(int) (Math.sin(54*Math.PI/180)*(pContribFromLibrary[3]*pSizeOfMovement))
    -
    (int) (Math.sin(18*Math.PI/180)*(pContribFromLibrary[4]*pSizeOfMovement));
    xEContig=xCircle

    +(int) (Math.cos(18*Math.PI/180)*(eContribFromLibrary[1]*eSizeOfMovement))
    +(int) (Math.cos(54*Math.PI/180)*(eContribFromLibrary[2]*eSizeOfMovement))
    -
    (int) (Math.cos(54*Math.PI/180)*(eContribFromLibrary[3]*eSizeOfMovement))
    -
    (int) (Math.cos(18*Math.PI/180)*(eContribFromLibrary[4]*eSizeOfMovement));
    yEContig=yCircle
        -(int) (eContribFromLibrary[0]*eSizeOfMovement)
        -
    (int) (Math.sin(18*Math.PI/180)*(eContribFromLibrary[1]*eSizeOfMovement))
    +(int) (Math.sin(54*Math.PI/180)*(eContribFromLibrary[2]*eSizeOfMovement))
    +(int) (Math.sin(54*Math.PI/180)*(eContribFromLibrary[3]*eSizeOfMovement))
    -
    (int) (Math.sin(18*Math.PI/180)*(eContribFromLibrary[4]*eSizeOfMovement));
    xWContig=xCircle

```

```

+ (int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
+ (int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement));
    yWContig=yCircle
        - (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement));
    } // end else if numberOfLibraries is 5

    // if numberOfLibraries is 6, 60 degrees per slice
    else if ( 6 == numberOfLibraries)
    {
        xPContig=xCircle
+ (int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
+ (int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement));
        yPContig=yCircle
            - (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
        + (int) (pContribFromLibrary [3] *pSizeOfMovement)
+ (int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement));
        xEContig=xCircle
+ (int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
+ (int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement));
        yEContig=yCircle
            - (int) (eContribFromLibrary [0] *eSizeOfMovement)

```

```

-
(int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (eContribFromLibrary[3] * eSizeOfMovement)

+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement));
xWContig=xCircle

+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))

-
(int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement));
yWContig=yCircle
- (int) (wContribFromLibrary[0] * wSizeOfMovement)
-
(int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (wContribFromLibrary[3] * wSizeOfMovement)

+ (int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement));
} // end 6

else if ( 7 == numberOfLibraries)
{
xPContig=xCircle

+ (int) (Math.cos(39*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(12*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.cos(63*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
-
(int) (Math.cos(15*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
-
(int) (Math.cos(36*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement));
yPContig=yCircle
- (int) (pContribFromLibrary[0] * pSizeOfMovement)
-
(int) (Math.sin(39*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.sin(12*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.sin(63*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (Math.sin(66*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))

```

```

+ (int) (Math.sin(15*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
-
(int) (Math.sin(36*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement));
    xEContig=xCircle
+ (int) (Math.cos(39*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(12*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(63*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
-
(int) (Math.cos(15*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
-
(int) (Math.cos(36*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement));
    yEContig=yCircle
        - (int) (eContribFromLibrary[0] * eSizeOfMovement)
-
(int) (Math.sin(39*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.sin(12*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.sin(63*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.sin(66*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.sin(15*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
-
(int) (Math.sin(36*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement));
    xWContig=xCircle
+ (int) (Math.cos(39*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(12*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(63*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
-
(int) (Math.cos(15*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
-
(int) (Math.cos(36*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement));
    yWContig=yCircle
        - (int) (wContribFromLibrary[0] * wSizeOfMovement)
-
(int) (Math.sin(39*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.sin(12*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.sin(63*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.sin(66*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.sin(15*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))

```

```

-
(int) (Math.sin(36*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement));
} // end 7

// if numberOfLibraries is 8, 45 degree pies
else if ( 8 == numberOfLibraries)
{
    xPContig=xCircle

+ (int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (pContribFromLibrary[2] * pSizeOfMovement)

+ (int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))

-
(int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
- (int) (pContribFromLibrary[6] * pSizeOfMovement)
-
(int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement));
    yPContig=yCircle
- (int) (pContribFromLibrary[0] * pSizeOfMovement)
-
(int) (Math.sin(45*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))

+ (int) (Math.sin(45*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (pContribFromLibrary[4] * pSizeOfMovement)

+ (int) (Math.sin(45*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))

-
(int) (Math.sin(45*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement));
    xEContig=xCircle

+ (int) (Math.cos(45*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (eContribFromLibrary[2] * eSizeOfMovement)

+ (int) (Math.cos(45*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))

-
(int) (Math.cos(45*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
- (int) (eContribFromLibrary[6] * eSizeOfMovement)
-
(int) (Math.cos(45*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement));
    yEContig=yCircle
- (int) (eContribFromLibrary[0] * eSizeOfMovement)
-
(int) (Math.sin(45*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))

+ (int) (Math.sin(45*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (eContribFromLibrary[4] * eSizeOfMovement)

+ (int) (Math.sin(45*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))

-
(int) (Math.sin(45*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement));

```

```

        xWContig=xCircle

+ (int) (Math.cos (45*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
        + (int) (wContribFromLibrary [2] *wSizeOfMovement)

+ (int) (Math.cos (45*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))

-
(int) (Math.cos (45*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
- (int) (wContribFromLibrary [6] *wSizeOfMovement)
-
(int) (Math.cos (45*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement));
        yWContig=yCircle
        - (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (45*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))

+ (int) (Math.sin (45*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
        + (int) (wContribFromLibrary [4] *wSizeOfMovement)

+ (int) (Math.sin (45*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))

-
(int) (Math.sin (45*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement));
    } // end 8

    // if numberOfLibraries is 9, 40 degree pies
    else if ( 9 == numberOfLibraries)
    {
        xPContig=xCircle

+ (int) (Math.cos (50*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
+ (int) (Math.cos (10*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
+ (int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
+ (int) (Math.cos (70*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
-
(int) (Math.cos (70*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement));
        yPContig=yCircle
        - (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (50*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (10*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
+ (int) (Math.sin (70*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))

```

```

+ (int) (Math.sin(70*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
-
(int) (Math.sin(10*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
-
(int) (Math.sin(50*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement));
    xEContig=xCircle

+ (int) (Math.cos(50*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(10*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(70*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
-
(int) (Math.cos(70*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
-
(int) (Math.cos(10*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
-
(int) (Math.cos(50*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement));
    yEContig=yCircle
        - (int) (eContribFromLibrary[0] * eSizeOfMovement)
-
(int) (Math.sin(50*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
-
(int) (Math.sin(10*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.sin(70*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.sin(70*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
-
(int) (Math.sin(10*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
-
(int) (Math.sin(50*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement));
    xWContig=xCircle

+ (int) (Math.cos(50*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(10*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(70*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
-
(int) (Math.cos(70*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))

```

```

-
(int) (Math.cos (10*Math.PI/180) * (wContribFromLibrary[7] *wSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (wContribFromLibrary[8] *wSizeOfMovement));
    yWContig=yCircle
        - (int) (wContribFromLibrary[0] *wSizeOfMovement)
-
(int) (Math.sin (50*Math.PI/180) * (wContribFromLibrary[1] *wSizeOfMovement))
-
(int) (Math.sin (10*Math.PI/180) * (wContribFromLibrary[2] *wSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary[3] *wSizeOfMovement))
+ (int) (Math.sin (70*Math.PI/180) * (wContribFromLibrary[4] *wSizeOfMovement))
+ (int) (Math.sin (70*Math.PI/180) * (wContribFromLibrary[5] *wSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary[6] *wSizeOfMovement))
-
(int) (Math.sin (10*Math.PI/180) * (wContribFromLibrary[7] *wSizeOfMovement))
-
(int) (Math.sin (50*Math.PI/180) * (wContribFromLibrary[8] *wSizeOfMovement));
    } // end 9

    // if numberOfLibraries is 10, 36 degree pies
    else if ( 10 == numberOfLibraries)
    {
        xPContig=xCircle
+ (int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary[1] *pSizeOfMovement))
+ (int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary[2] *pSizeOfMovement))
+ (int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary[3] *pSizeOfMovement))
+ (int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary[4] *pSizeOfMovement))

-
(int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary[6] *pSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary[7] *pSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary[8] *pSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary[9] *pSizeOfMovement));
        yPContig=yCircle
            - (int) (pContribFromLibrary[0] *pSizeOfMovement)
-
(int) (Math.sin (54*Math.PI/180) * (pContribFromLibrary[1] *pSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (pContribFromLibrary[2] *pSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (pContribFromLibrary[3] *pSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (pContribFromLibrary[4] *pSizeOfMovement))
        + (int) (pContribFromLibrary[5] *pSizeOfMovement)

```

```

+ (int) (Math.sin (54*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
-
(int) (Math.sin (54*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement));
    xEContig=xCircle

+ (int) (Math.cos (54*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
+ (int) (Math.cos (18*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
+ (int) (Math.cos (18*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
+ (int) (Math.cos (54*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))

-
(int) (Math.cos (54*Math.PI/180) * (eContribFromLibrary [6] *eSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement));
    yEContig=yCircle
        - (int) (eContribFromLibrary [0] *eSizeOfMovement)
-
(int) (Math.sin (54*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
    + (int) (eContribFromLibrary [5] *eSizeOfMovement)

+ (int) (Math.sin (54*Math.PI/180) * (eContribFromLibrary [6] *eSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
-
(int) (Math.sin (54*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement));
    xWContig=xCircle

+ (int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
+ (int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
+ (int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
+ (int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))

-
(int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))

```

```

-
(int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement));
    yWContig=yCircle
        - (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
    + (int) (wContribFromLibrary [5] *wSizeOfMovement)
+ (int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
-
(int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement));
    } // end 10

    else if ( 11 == numberOfLibraries)
    {
        xPContig=xCircle
+ (int) (Math.cos (60*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
+ (int) (Math.cos (27*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
+ (int) (Math.cos (6*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
+ (int) (Math.cos (39*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
+ (int) (Math.cos (72*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
-
(int) (Math.cos (75*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
-
(int) (Math.cos (9*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
-
(int) (Math.cos (24*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
-
(int) (Math.cos (57*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement));
        yPContig=yCircle
            - (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (60*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (27*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))

```

```

+ (int) (Math.sin(6*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (Math.sin(39*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
+ (int) (Math.sin(72*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
+ (int) (Math.sin(75*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
+ (int) (Math.sin(42*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
+ (int) (Math.sin(9*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))
-
(int) (Math.sin(24*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
-
(int) (Math.sin(57*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement));
    xEContig=xCircle

+ (int) (Math.cos(60*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(27*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(39*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.cos(72*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
-
(int) (Math.cos(75*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
-
(int) (Math.cos(42*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
-
(int) (Math.cos(9*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
-
(int) (Math.cos(24*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
-
(int) (Math.cos(57*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement));
    yEContig=yCircle
    - (int) (eContribFromLibrary[0] * eSizeOfMovement)
-
(int) (Math.sin(60*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
-
(int) (Math.sin(27*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.sin(39*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.sin(72*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
+ (int) (Math.sin(75*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.sin(42*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.sin(9*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
-
(int) (Math.sin(24*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))

```

```

-
(int) (Math.sin(57*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement));
    xWContig=xCircle

+ (int) (Math.cos(60*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(27*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(39*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(72*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
-
(int) (Math.cos(75*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
-
(int) (Math.cos(42*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
-
(int) (Math.cos(9*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
-
(int) (Math.cos(24*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
-
(int) (Math.cos(57*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement));
    yWContig=yCircle
        - (int) (wContribFromLibrary[0] * wSizeOfMovement)
-
(int) (Math.sin(60*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
-
(int) (Math.sin(27*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.sin(39*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.sin(72*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
+ (int) (Math.sin(75*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.sin(42*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.sin(9*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
-
(int) (Math.sin(24*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
-
(int) (Math.sin(57*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement));
    } // end 11

    // if numberOfLibraries is 12, 30 degree pies
    else if ( 12 == numberOfLibraries)
    {
        xPContig=xCircle

+ (int) (Math.cos(60*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
        + (int) (pContribFromLibrary[3] * pSizeOfMovement)

```

```

+ (int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [4] * pSizeOfMovement))
+ (int) (Math.cos (60*Math.PI/180) * (pContribFromLibrary [5] * pSizeOfMovement))

-
(int) (Math.cos (60*Math.PI/180) * (pContribFromLibrary [7] * pSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [8] * pSizeOfMovement))
- (int) (pContribFromLibrary [9] * pSizeOfMovement)
-
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [10] * pSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (pContribFromLibrary [11] * pSizeOfMovement));
    yPContig=yCircle
        - (int) (pContribFromLibrary [0] * pSizeOfMovement)
-
(int) (Math.sin (60*Math.PI/180) * (pContribFromLibrary [1] * pSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [2] * pSizeOfMovement))

+ (int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [4] * pSizeOfMovement))
+ (int) (Math.sin (60*Math.PI/180) * (pContribFromLibrary [5] * pSizeOfMovement))
    + (int) (pContribFromLibrary [6] * pSizeOfMovement)

+ (int) (Math.sin (60*Math.PI/180) * (pContribFromLibrary [7] * pSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [8] * pSizeOfMovement))

-
(int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [10] * pSizeOfMovement))
-
(int) (Math.sin (60*Math.PI/180) * (pContribFromLibrary [11] * pSizeOfMovement));
    xEContig=xCircle

+ (int) (Math.cos (60*Math.PI/180) * (eContribFromLibrary [1] * eSizeOfMovement))
+ (int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [2] * eSizeOfMovement))
    + (int) (eContribFromLibrary [3] * eSizeOfMovement)

+ (int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [4] * eSizeOfMovement))
+ (int) (Math.cos (60*Math.PI/180) * (eContribFromLibrary [5] * eSizeOfMovement))

-
(int) (Math.cos (60*Math.PI/180) * (eContribFromLibrary [7] * eSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [8] * eSizeOfMovement))
    - (int) (eContribFromLibrary [9] * eSizeOfMovement)
-
(int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [10] * eSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (eContribFromLibrary [11] * eSizeOfMovement));
    yEContig=yCircle
        - (int) (eContribFromLibrary [0] * eSizeOfMovement)

```

```

-
(int) (Math.sin(60*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))

+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.sin(60*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
+ (int) (eContribFromLibrary[6] * eSizeOfMovement)
+ (int) (Math.sin(60*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))

-
(int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
-
(int) (Math.sin(60*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement));
xWContig=xCircle
+ (int) (Math.cos(60*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (wContribFromLibrary[3] * wSizeOfMovement)
+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(60*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))

-
(int) (Math.cos(60*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
- (int) (wContribFromLibrary[9] * wSizeOfMovement)
-
(int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
-
(int) (Math.cos(60*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement));
yWContig=yCircle
- (int) (wContribFromLibrary[0] * wSizeOfMovement)
-
(int) (Math.sin(60*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))

+ (int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.sin(60*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
+ (int) (wContribFromLibrary[6] * wSizeOfMovement)
+ (int) (Math.sin(60*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))

```

```

-
(int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
-
(int) (Math.sin(60*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement));
    } // end 12

    // if numberOfLibraries is 13, 28 & one 24-degree pies
    else if ( 13 == numberOfLibraries)
    {
        xPContig=xCircle
+ (int) (Math.cos(62*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(34*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (Math.cos(22*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
+ (int) (Math.cos(50*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
-
(int) (Math.cos(74*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
-
(int) (Math.cos(46*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))
-
(int) (Math.cos(18*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
-
(int) (Math.cos(10*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))
-
(int) (Math.cos(38*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement));
        yPContig=yCircle
            - (int) (pContribFromLibrary[0] * pSizeOfMovement)
            -
(int) (Math.sin(62*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
            -
(int) (Math.sin(34*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
            -
(int) (Math.sin(6*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (Math.sin(22*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
+ (int) (Math.sin(50*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
+ (int) (Math.sin(74*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
+ (int) (Math.sin(46*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))
+ (int) (Math.sin(18*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
-
(int) (Math.sin(10*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))

```

```

-
(int) (Math.sin(38*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement));
    xEContig=xCircle
+ (int) (Math.cos(62*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(34*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(22*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.cos(50*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
-
(int) (Math.cos(74*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
-
(int) (Math.cos(46*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
-
(int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
-
(int) (Math.cos(10*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
-
(int) (Math.cos(38*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement));
    yEContig=yCircle
        - (int) (eContribFromLibrary[0] * eSizeOfMovement)
-
(int) (Math.sin(62*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
-
(int) (Math.sin(34*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
-
(int) (Math.sin(6*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.sin(22*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.sin(50*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.sin(74*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.sin(46*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
+ (int) (Math.sin(18*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
-
(int) (Math.sin(10*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
-
(int) (Math.sin(38*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement));
    xWContig=xCircle

```

```

+ (int) (Math.cos (62*Math.PI/180) * (wContribFromLibrary[1] *wSizeOfMovement))
+ (int) (Math.cos (34*Math.PI/180) * (wContribFromLibrary[2] *wSizeOfMovement))
+ (int) (Math.cos (6*Math.PI/180) * (wContribFromLibrary[3] *wSizeOfMovement))
+ (int) (Math.cos (22*Math.PI/180) * (wContribFromLibrary[4] *wSizeOfMovement))
+ (int) (Math.cos (50*Math.PI/180) * (wContribFromLibrary[5] *wSizeOfMovement))
+ (int) (Math.cos (78*Math.PI/180) * (wContribFromLibrary[6] *wSizeOfMovement))
-
(int) (Math.cos (74*Math.PI/180) * (wContribFromLibrary[7] *wSizeOfMovement))
-
(int) (Math.cos (46*Math.PI/180) * (wContribFromLibrary[8] *wSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary[9] *wSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (wContribFromLibrary[10] *wSizeOfMovement))
-
(int) (Math.cos (38*Math.PI/180) * (wContribFromLibrary[11] *wSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (wContribFromLibrary[12] *wSizeOfMovement));
    yWContig=yCircle
        - (int) (wContribFromLibrary[0] *wSizeOfMovement)
        -
(int) (Math.sin (62*Math.PI/180) * (wContribFromLibrary[1] *wSizeOfMovement))
-
(int) (Math.sin (34*Math.PI/180) * (wContribFromLibrary[2] *wSizeOfMovement))
-
(int) (Math.sin (6*Math.PI/180) * (wContribFromLibrary[3] *wSizeOfMovement))
+ (int) (Math.sin (22*Math.PI/180) * (wContribFromLibrary[4] *wSizeOfMovement))
+ (int) (Math.sin (50*Math.PI/180) * (wContribFromLibrary[5] *wSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (wContribFromLibrary[6] *wSizeOfMovement))
+ (int) (Math.sin (74*Math.PI/180) * (wContribFromLibrary[7] *wSizeOfMovement))
+ (int) (Math.sin (46*Math.PI/180) * (wContribFromLibrary[8] *wSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary[9] *wSizeOfMovement))
-
(int) (Math.sin (10*Math.PI/180) * (wContribFromLibrary[10] *wSizeOfMovement))
-
(int) (Math.sin (38*Math.PI/180) * (wContribFromLibrary[11] *wSizeOfMovement))
-
(int) (Math.sin (66*Math.PI/180) * (wContribFromLibrary[12] *wSizeOfMovement));
    } // end 13

    // if numberOfLibraries is 14, 26 & one 22-degree pies
    else if ( 14 == numberOfLibraries)
    {
        xPContig=xCircle

```

```

+ (int) (Math.cos (64*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
+ (int) (Math.cos (38*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
+ (int) (Math.cos (12*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
+ (int) (Math.cos (14*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
+ (int) (Math.cos (40*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
+ (int) (Math.cos (66*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
-
(int) (Math.cos (88*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
-
(int) (Math.cos (62*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
-
(int) (Math.cos (16*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
-
(int) (Math.cos (68*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement));
    yPContig=yCircle
        - (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (64*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (38*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
-
(int) (Math.sin (12*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
+ (int) (Math.sin (14*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
+ (int) (Math.sin (40*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.sin (88*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.sin (62*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
+ (int) (Math.sin (36*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
+ (int) (Math.sin (10*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
-
(int) (Math.sin (16*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
-
(int) (Math.sin (42*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
-
(int) (Math.sin (68*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement));
    xEContig=xCircle
+ (int) (Math.cos (64*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))

```

```

+ (int) (Math.cos (38*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
+ (int) (Math.cos (12*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
+ (int) (Math.cos (14*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
+ (int) (Math.cos (40*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))
+ (int) (Math.cos (66*Math.PI/180) * (eContribFromLibrary [6] *eSizeOfMovement))
-
(int) (Math.cos (88*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
-
(int) (Math.cos (62*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (eContribFromLibrary [10] *eSizeOfMovement))
-
(int) (Math.cos (16*Math.PI/180) * (eContribFromLibrary [11] *eSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (eContribFromLibrary [12] *eSizeOfMovement))
-
(int) (Math.cos (68*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement));
    yEContig=yCircle
        - (int) (eContribFromLibrary [0] *eSizeOfMovement)
-
(int) (Math.sin (64*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
-
(int) (Math.sin (38*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
-
(int) (Math.sin (12*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
+ (int) (Math.sin (14*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
+ (int) (Math.sin (40*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (eContribFromLibrary [6] *eSizeOfMovement))
+ (int) (Math.sin (88*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
+ (int) (Math.sin (62*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
+ (int) (Math.sin (36*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))
+ (int) (Math.sin (10*Math.PI/180) * (eContribFromLibrary [10] *eSizeOfMovement))
-
(int) (Math.sin (16*Math.PI/180) * (eContribFromLibrary [11] *eSizeOfMovement))
-
(int) (Math.sin (42*Math.PI/180) * (eContribFromLibrary [12] *eSizeOfMovement))
-
(int) (Math.sin (68*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement));
    xWContig=xCircle
+ (int) (Math.cos (64*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))

```

```

+ (int) (Math.cos (38*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
+ (int) (Math.cos (12*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
+ (int) (Math.cos (14*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
+ (int) (Math.cos (40*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
+ (int) (Math.cos (66*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
-
(int) (Math.cos (88*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
-
(int) (Math.cos (62*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
-
(int) (Math.cos (16*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
-
(int) (Math.cos (68*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement));
    yWContig=yCircle
        - (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (64*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
-
(int) (Math.sin (38*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
-
(int) (Math.sin (12*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
+ (int) (Math.sin (14*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
+ (int) (Math.sin (40*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
+ (int) (Math.sin (88*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
+ (int) (Math.sin (62*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
+ (int) (Math.sin (36*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
+ (int) (Math.sin (10*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
-
(int) (Math.sin (16*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
-
(int) (Math.sin (42*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
-
(int) (Math.sin (68*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement));
    } // end 14

    // if numberOfLibraries is 15, 24 degree pies
    else if ( 15 == numberOfLibraries)
    {

```

```

        xPContig=xCircle
+ (int) (Math.cos(66*Math.PI/180) * (pContribFromLibrary[1] *pSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (pContribFromLibrary[2] *pSizeOfMovement))
+ (int) (Math.cos(18*Math.PI/180) * (pContribFromLibrary[3] *pSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (pContribFromLibrary[4] *pSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[5] *pSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (pContribFromLibrary[6] *pSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (pContribFromLibrary[7] *pSizeOfMovement))
-
(int) (Math.cos(78*Math.PI/180) * (pContribFromLibrary[8] *pSizeOfMovement))
-
(int) (Math.cos(54*Math.PI/180) * (pContribFromLibrary[9] *pSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[10] *pSizeOfMovement))
-
(int) (Math.cos(6*Math.PI/180) * (pContribFromLibrary[11] *pSizeOfMovement))
-
(int) (Math.cos(18*Math.PI/180) * (pContribFromLibrary[12] *pSizeOfMovement))
-
(int) (Math.cos(42*Math.PI/180) * (pContribFromLibrary[13] *pSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (pContribFromLibrary[14] *pSizeOfMovement));
        yPContig=yCircle
- (int) (pContribFromLibrary[0] *pSizeOfMovement)
-
(int) (Math.sin(66*Math.PI/180) * (pContribFromLibrary[1] *pSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (pContribFromLibrary[2] *pSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (pContribFromLibrary[3] *pSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (pContribFromLibrary[4] *pSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[5] *pSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (pContribFromLibrary[6] *pSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (pContribFromLibrary[7] *pSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (pContribFromLibrary[8] *pSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (pContribFromLibrary[9] *pSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[10] *pSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (pContribFromLibrary[11] *pSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (pContribFromLibrary[12] *pSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (pContribFromLibrary[13] *pSizeOfMovement))

```

```

(int) (Math.sin(66*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement));
    xEContig=xCircle
+ (int) (Math.cos(66*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
-
(int) (Math.cos(78*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
-
(int) (Math.cos(54*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
-
(int) (Math.cos(6*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
-
(int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
-
(int) (Math.cos(42*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement));
    yEContig=yCircle
        - (int) (eContribFromLibrary[0] * eSizeOfMovement)
-
(int) (Math.sin(66*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))

```

```

(int) (Math.sin(42*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement));
    xWContig=xCircle

+ (int) (Math.cos(66*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(18*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
-
(int) (Math.cos(78*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
-
(int) (Math.cos(54*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
-
(int) (Math.cos(6*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
-
(int) (Math.cos(18*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
-
(int) (Math.cos(42*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement));
    yWContig=yCircle
        - (int) (wContribFromLibrary[0] * wSizeOfMovement)
        -
(int) (Math.sin(66*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))

```

```

-
(int) (Math.sin(18*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement));
    } // end 15
} // end calcXY_1to15()
//*****
void calcXY_16to21()
{
    // if numberOfLibraries is 16, alternate 22/23 degree pies
    if ( 16 == numberOfLibraries)
    {
        xPContig=xCircle

+ (int) (Math.cos(68*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.cos(23*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
        + (int) (pContribFromLibrary[4] * pSizeOfMovement)
+ (int) (Math.cos(22*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
+ (int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
+ (int) (Math.cos(67*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))

-
(int) (Math.cos(68*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
-
(int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))
-
(int) (Math.cos(23*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
        - (int) (pContribFromLibrary[12] * pSizeOfMovement)
-
(int) (Math.cos(22*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
-
(int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))
-
(int) (Math.cos(67*Math.PI/180) * (pContribFromLibrary[15] * pSizeOfMovement));
        yPContig=yCircle
        - (int) (pContribFromLibrary[0] * pSizeOfMovement)
-
(int) (Math.sin(68*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
-
(int) (Math.sin(45*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
-
(int) (Math.sin(23*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))

+ (int) (Math.sin(22*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
+ (int) (Math.sin(45*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
+ (int) (Math.sin(67*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))

```

```

        +(int) (pContribFromLibrary[8] *pSizeOfMovement)
+(int) (Math.sin(68*Math.PI/180) * (pContribFromLibrary[9] *pSizeOfMovement))
+(int) (Math.sin(45*Math.PI/180) * (pContribFromLibrary[10] *pSizeOfMovement))
+(int) (Math.sin(23*Math.PI/180) * (pContribFromLibrary[11] *pSizeOfMovement))
-
(int) (Math.sin(22*Math.PI/180) * (pContribFromLibrary[13] *pSizeOfMovement))
-
(int) (Math.sin(45*Math.PI/180) * (pContribFromLibrary[14] *pSizeOfMovement))
-
(int) (Math.sin(67*Math.PI/180) * (pContribFromLibrary[15] *pSizeOfMovement));
    xEContig=xCircle
+(int) (Math.cos(68*Math.PI/180) * (eContribFromLibrary[1] *eSizeOfMovement))
+(int) (Math.cos(45*Math.PI/180) * (eContribFromLibrary[2] *eSizeOfMovement))
+(int) (Math.cos(23*Math.PI/180) * (eContribFromLibrary[3] *eSizeOfMovement))
    +(int) (eContribFromLibrary[4] *eSizeOfMovement)
+(int) (Math.cos(22*Math.PI/180) * (eContribFromLibrary[5] *eSizeOfMovement))
+(int) (Math.cos(45*Math.PI/180) * (eContribFromLibrary[6] *eSizeOfMovement))
+(int) (Math.cos(67*Math.PI/180) * (eContribFromLibrary[7] *eSizeOfMovement))
-
(int) (Math.cos(68*Math.PI/180) * (eContribFromLibrary[9] *eSizeOfMovement))
-
(int) (Math.cos(45*Math.PI/180) * (eContribFromLibrary[10] *eSizeOfMovement))
-
(int) (Math.cos(23*Math.PI/180) * (eContribFromLibrary[11] *eSizeOfMovement))
    -(int) (eContribFromLibrary[12] *eSizeOfMovement)
-
(int) (Math.cos(22*Math.PI/180) * (eContribFromLibrary[13] *eSizeOfMovement))
-
(int) (Math.cos(45*Math.PI/180) * (eContribFromLibrary[14] *eSizeOfMovement))
-
(int) (Math.cos(67*Math.PI/180) * (eContribFromLibrary[15] *eSizeOfMovement));
    yEContig=yCircle
        -(int) (eContribFromLibrary[0] *eSizeOfMovement)
-
(int) (Math.sin(68*Math.PI/180) * (eContribFromLibrary[1] *eSizeOfMovement))
-
(int) (Math.sin(45*Math.PI/180) * (eContribFromLibrary[2] *eSizeOfMovement))
-
(int) (Math.sin(23*Math.PI/180) * (eContribFromLibrary[3] *eSizeOfMovement))

+(int) (Math.sin(22*Math.PI/180) * (eContribFromLibrary[5] *eSizeOfMovement))
+(int) (Math.sin(45*Math.PI/180) * (eContribFromLibrary[6] *eSizeOfMovement))

```

```

+ (int) (Math.sin(67*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
      + (int) (eContribFromLibrary[8] * eSizeOfMovement)

+ (int) (Math.sin(68*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))

+ (int) (Math.sin(45*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))

+ (int) (Math.sin(23*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))

-
(int) (Math.sin(22*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
-
(int) (Math.sin(45*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
-
(int) (Math.sin(67*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement));
    xWContig=xCircle

+ (int) (Math.cos(68*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))

+ (int) (Math.cos(45*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))

+ (int) (Math.cos(23*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
      + (int) (wContribFromLibrary[4] * wSizeOfMovement)

+ (int) (Math.cos(22*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))

+ (int) (Math.cos(45*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))

+ (int) (Math.cos(67*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))

-
(int) (Math.cos(68*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
-
(int) (Math.cos(45*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
-
(int) (Math.cos(23*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
      - (int) (wContribFromLibrary[12] * wSizeOfMovement)
-
(int) (Math.cos(22*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
-
(int) (Math.cos(45*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
-
(int) (Math.cos(67*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement));
    yWContig=yCircle
      - (int) (wContribFromLibrary[0] * wSizeOfMovement)
-
(int) (Math.sin(68*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
-
(int) (Math.sin(45*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
-
(int) (Math.sin(23*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))

+ (int) (Math.sin(22*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))

+ (int) (Math.sin(45*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))

```

```

+ (int) (Math.sin(67*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (wContribFromLibrary[8] * wSizeOfMovement)

+ (int) (Math.sin(68*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))

+ (int) (Math.sin(45*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))

+ (int) (Math.sin(23*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))

-
(int) (Math.sin(22*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
-
(int) (Math.sin(45*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
-
(int) (Math.sin(67*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement));
} // end 16

else if ( 17 == numberOfLibraries)
{
    xPContig=xCircle

+ (int) (Math.cos(69*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(48*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.cos(27*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
+ (int) (Math.cos(15*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
+ (int) (Math.cos(36*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
+ (int) (Math.cos(57*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))
-
(int) (Math.cos(81*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
-
(int) (Math.cos(60*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))
-
(int) (Math.cos(39*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
-
(int) (Math.cos(18*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement))
-
(int) (Math.cos(3*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
-
(int) (Math.cos(24*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))
-
(int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[15] * pSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (pContribFromLibrary[16] * pSizeOfMovement));
    yPContig=yCircle
- (int) (pContribFromLibrary[0] * pSizeOfMovement)
-
(int) (Math.sin(69*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))

```

```

-
(int) (Math.sin(48*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
-
(int) (Math.sin(27*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
-
(int) (Math.sin(6*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
+ (int) (Math.sin(15*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
+ (int) (Math.sin(36*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
+ (int) (Math.sin(57*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))
+ (int) (Math.sin(81*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
+ (int) (Math.sin(60*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))
+ (int) (Math.sin(39*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
+ (int) (Math.sin(18*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement))
-
(int) (Math.sin(3*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
-
(int) (Math.sin(24*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))
-
(int) (Math.sin(45*Math.PI/180) * (pContribFromLibrary[15] * pSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (pContribFromLibrary[16] * pSizeOfMovement));

```

xEContig=xCircle

```

+ (int) (Math.cos(69*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(48*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(27*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.cos(15*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
+ (int) (Math.cos(36*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.cos(57*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
-
(int) (Math.cos(81*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
-
(int) (Math.cos(60*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
-
(int) (Math.cos(39*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
-
(int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))

```

```

-
(int) (Math.cos (3*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement))
-
(int) (Math.cos (24*Math.PI/180) * (eContribFromLibrary [14] *eSizeOfMovement))
-
(int) (Math.cos (45*Math.PI/180) * (eContribFromLibrary [15] *eSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (eContribFromLibrary [16] *eSizeOfMovement));
    yEContig=yCircle
        - (int) (eContribFromLibrary [0] *eSizeOfMovement)
-
(int) (Math.sin (69*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
-
(int) (Math.sin (48*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
-
(int) (Math.sin (27*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
-
(int) (Math.sin (6*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
+ (int) (Math.sin (15*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))
+ (int) (Math.sin (36*Math.PI/180) * (eContribFromLibrary [6] *eSizeOfMovement))
+ (int) (Math.sin (57*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
+ (int) (Math.sin (81*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))
+ (int) (Math.sin (60*Math.PI/180) * (eContribFromLibrary [10] *eSizeOfMovement))
+ (int) (Math.sin (39*Math.PI/180) * (eContribFromLibrary [11] *eSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (eContribFromLibrary [12] *eSizeOfMovement))
-
(int) (Math.sin (3*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement))
-
(int) (Math.sin (24*Math.PI/180) * (eContribFromLibrary [14] *eSizeOfMovement))
-
(int) (Math.sin (45*Math.PI/180) * (eContribFromLibrary [15] *eSizeOfMovement))
-
(int) (Math.sin (66*Math.PI/180) * (eContribFromLibrary [16] *eSizeOfMovement));
    xWContig=xCircle
+ (int) (Math.cos (69*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
+ (int) (Math.cos (48*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
+ (int) (Math.cos (27*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
+ (int) (Math.cos (6*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
+ (int) (Math.cos (15*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
+ (int) (Math.cos (36*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
+ (int) (Math.cos (57*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))

```

```

+ (int) (Math.cos (78*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
-
(int) (Math.cos (81*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
-
(int) (Math.cos (39*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
-
(int) (Math.cos (3*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
-
(int) (Math.cos (24*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
-
(int) (Math.cos (45*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement));
    yWContig=yCircle
        - (int) (wContribFromLibrary [0] *wSizeOfMovement)
        -
(int) (Math.sin (69*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
-
(int) (Math.sin (48*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
-
(int) (Math.sin (27*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
-
(int) (Math.sin (6*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
+ (int) (Math.sin (15*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
+ (int) (Math.sin (36*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
+ (int) (Math.sin (57*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
+ (int) (Math.sin (81*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
+ (int) (Math.sin (60*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
+ (int) (Math.sin (39*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
-
(int) (Math.sin (3*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
-
(int) (Math.sin (24*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
-
(int) (Math.sin (45*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
-
(int) (Math.sin (66*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement));
    } // end 17

    // if numberOfLibraries is 18, 20-degree pies
    else if ( 18 == numberOfLibraries)
    {

```

```

xPContig=xCircle
+ (int) (Math.cos (70*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
+ (int) (Math.cos (50*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
+ (int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
+ (int) (Math.cos (10*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
+ (int) (Math.cos (10*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
+ (int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.cos (50*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.cos (70*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))

-
(int) (Math.cos (70*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (pContribFromLibrary [14] *pSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))
-
(int) (Math.cos (70*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement));

yPContig=yCircle
- (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (70*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (50*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
-
(int) (Math.sin (10*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
+ (int) (Math.sin (10*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.sin (50*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.sin (70*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
+ (int) (pContribFromLibrary [9] *pSizeOfMovement)
+ (int) (Math.sin (70*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))

```

```

+ (int) (Math.sin(50*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement))
+ (int) (Math.sin(10*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
-
(int) (Math.sin(10*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[15] * pSizeOfMovement))
-
(int) (Math.sin(50*Math.PI/180) * (pContribFromLibrary[16] * pSizeOfMovement))
-
(int) (Math.sin(70*Math.PI/180) * (pContribFromLibrary[17] * pSizeOfMovement));

```

xEContig=xCircle

```

+ (int) (Math.cos(70*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(50*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(10*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.cos(10*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.cos(50*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.cos(70*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
-
(int) (Math.cos(70*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
-
(int) (Math.cos(50*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
-
(int) (Math.cos(10*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
-
(int) (Math.cos(10*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
-
(int) (Math.cos(50*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
-
(int) (Math.cos(70*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement));

```

yEContig=yCircle

```

- (int) (eContribFromLibrary[0] * eSizeOfMovement)
-
(int) (Math.sin(70*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
-
(int) (Math.sin(50*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))

```

```

-
(int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
-
(int) (Math.sin(10*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.sin(10*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.sin(50*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.sin(70*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
+ (int) (eContribFromLibrary[9] * eSizeOfMovement)
+ (int) (Math.sin(70*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
+ (int) (Math.sin(50*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
+ (int) (Math.sin(10*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
-
(int) (Math.sin(10*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
-
(int) (Math.sin(50*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
-
(int) (Math.sin(70*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement));

```

xWContig=xCircle

```

+ (int) (Math.cos(70*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(50*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(10*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(10*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.cos(50*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.cos(70*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
-
(int) (Math.cos(70*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
-
(int) (Math.cos(50*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
-
(int) (Math.cos(10*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))

```

```

-
(int) (Math.cos (10*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))
-
(int) (Math.cos (70*Math.PI/180) * (wContribFromLibrary [17] *wSizeOfMovement));

    yWContig=yCircle
        - (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (70*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
-
(int) (Math.sin (50*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
-
(int) (Math.sin (10*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
+ (int) (Math.sin (10*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
+ (int) (Math.sin (50*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
+ (int) (Math.sin (70*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
    + (int) (wContribFromLibrary [9] *wSizeOfMovement)
+ (int) (Math.sin (70*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
+ (int) (Math.sin (50*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
+ (int) (Math.sin (10*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
-
(int) (Math.sin (10*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
-
(int) (Math.sin (50*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))
-
(int) (Math.sin (70*Math.PI/180) * (wContribFromLibrary [17] *wSizeOfMovement));
    } // end 18

    // if numberOfLibraries is 19, 19-degree pies plus one 18-degree
    else if ( 19 == numberOfLibraries)
    {
        xPContig=xCircle
+ (int) (Math.cos (71*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
+ (int) (Math.cos (52*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
+ (int) (Math.cos (33*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))

```

```

+ (int) (Math.cos (14*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
+ (int) (Math.cos (5*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
+ (int) (Math.cos (24*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.cos (43*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.cos (62*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
    + (int) (Math.cos (81*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
-
(int) (Math.cos (80*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
-
(int) (Math.cos (61*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
-
(int) (Math.cos (23*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
-
(int) (Math.cos (4*Math.PI/180) * (pContribFromLibrary [14] *pSizeOfMovement))
-
(int) (Math.cos (15*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))
-
(int) (Math.cos (34*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))
-
(int) (Math.cos (53*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement))
-
(int) (Math.cos (72*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement));

yPContig=yCircle
    - (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (71*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (52*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
-
(int) (Math.sin (33*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
-
(int) (Math.sin (14*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
+ (int) (Math.sin (5*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
+ (int) (Math.sin (24*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.sin (43*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.sin (62*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
+ (int) (Math.sin (81*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
+ (int) (Math.sin (80*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
+ (int) (Math.sin (61*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))

```

```

+ (int) (Math.sin (23*Math.PI/180) * (pContribFromLibrary [13] * pSizeOfMovement))
+ (int) (Math.sin (4*Math.PI/180) * (pContribFromLibrary [14] * pSizeOfMovement))
-
(int) (Math.sin (15*Math.PI/180) * (pContribFromLibrary [15] * pSizeOfMovement))
-
(int) (Math.sin (34*Math.PI/180) * (pContribFromLibrary [16] * pSizeOfMovement))
-
(int) (Math.sin (53*Math.PI/180) * (pContribFromLibrary [17] * pSizeOfMovement))
-
(int) (Math.sin (72*Math.PI/180) * (pContribFromLibrary [18] * pSizeOfMovement));
    xEContig=xCircle

+ (int) (Math.cos (71*Math.PI/180) * (eContribFromLibrary [1] * eSizeOfMovement))
+ (int) (Math.cos (52*Math.PI/180) * (eContribFromLibrary [2] * eSizeOfMovement))
+ (int) (Math.cos (33*Math.PI/180) * (eContribFromLibrary [3] * eSizeOfMovement))
+ (int) (Math.cos (14*Math.PI/180) * (eContribFromLibrary [4] * eSizeOfMovement))
+ (int) (Math.cos (5*Math.PI/180) * (eContribFromLibrary [5] * eSizeOfMovement))
+ (int) (Math.cos (24*Math.PI/180) * (eContribFromLibrary [6] * eSizeOfMovement))
+ (int) (Math.cos (43*Math.PI/180) * (eContribFromLibrary [7] * eSizeOfMovement))
+ (int) (Math.cos (62*Math.PI/180) * (eContribFromLibrary [8] * eSizeOfMovement))
    + (int) (Math.cos (81*Math.PI/180) * (eContribFromLibrary [9] * eSizeOfMovement))
-
(int) (Math.cos (80*Math.PI/180) * (eContribFromLibrary [10] * eSizeOfMovement))
-
(int) (Math.cos (61*Math.PI/180) * (eContribFromLibrary [11] * eSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (eContribFromLibrary [12] * eSizeOfMovement))
-
(int) (Math.cos (23*Math.PI/180) * (eContribFromLibrary [13] * eSizeOfMovement))
-
(int) (Math.cos (4*Math.PI/180) * (eContribFromLibrary [14] * eSizeOfMovement))
-
(int) (Math.cos (15*Math.PI/180) * (eContribFromLibrary [15] * eSizeOfMovement))
-
(int) (Math.cos (34*Math.PI/180) * (eContribFromLibrary [16] * eSizeOfMovement))
-
(int) (Math.cos (53*Math.PI/180) * (eContribFromLibrary [17] * eSizeOfMovement))
-
(int) (Math.cos (72*Math.PI/180) * (eContribFromLibrary [18] * eSizeOfMovement));

    yEContig=yCircle
        - (int) (eContribFromLibrary [0] * eSizeOfMovement)
-
(int) (Math.sin (71*Math.PI/180) * (eContribFromLibrary [1] * eSizeOfMovement))
-
(int) (Math.sin (52*Math.PI/180) * (eContribFromLibrary [2] * eSizeOfMovement))

```

```

-
(int) (Math.sin(33*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
-
(int) (Math.sin(14*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.sin(5*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
+ (int) (Math.sin(24*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.sin(43*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.sin(62*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
+ (int) (Math.sin(81*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
+ (int) (Math.sin(80*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
+ (int) (Math.sin(61*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.sin(42*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
+ (int) (Math.sin(23*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
+ (int) (Math.sin(4*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
-
(int) (Math.sin(15*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
-
(int) (Math.sin(34*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
-
(int) (Math.sin(53*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
-
(int) (Math.sin(72*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement));
    xWContig=xCircle

+ (int) (Math.cos(71*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(52*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(33*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(14*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(5*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
+ (int) (Math.cos(24*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.cos(43*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.cos(62*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
    + (int) (Math.cos(81*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
-
(int) (Math.cos(80*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
-
(int) (Math.cos(61*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
-
(int) (Math.cos(42*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))

```

```

-
(int) (Math.cos (23*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
-
(int) (Math.cos (4*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
-
(int) (Math.cos (15*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))
-
(int) (Math.cos (34*Math.PI/180) * (wContribFromLibrary[16] * wSizeOfMovement))
-
(int) (Math.cos (53*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
-
(int) (Math.cos (72*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement));

```

```

    yWContig=yCircle
        - (int) (wContribFromLibrary[0] * wSizeOfMovement)
-
(int) (Math.sin (71*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
-
(int) (Math.sin (52*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
-
(int) (Math.sin (33*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
-
(int) (Math.sin (14*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.sin (5*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
+ (int) (Math.sin (24*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.sin (43*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.sin (62*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
+ (int) (Math.sin (81*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
+ (int) (Math.sin (80*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
+ (int) (Math.sin (61*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
+ (int) (Math.sin (23*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
+ (int) (Math.sin (4*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
-
(int) (Math.sin (15*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))
-
(int) (Math.sin (34*Math.PI/180) * (wContribFromLibrary[16] * wSizeOfMovement))
-
(int) (Math.sin (53*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
-
(int) (Math.sin (72*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement));
} // end 19

```

```

// if numberOfLibraries is 20, 18-degree pies
else if ( 20 == numberOfLibraries)
{
    xPContig=xCircle

```

```

+ (int) (Math.cos (72*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
+ (int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
+ (int) (Math.cos (36*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
+ (int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
    + (int) (pContribFromLibrary [5] *pSizeOfMovement)
+ (int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.cos (36*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
    + (int) (Math.cos (72*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))

-
(int) (Math.cos (72*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary [14] *pSizeOfMovement))
    - (int) (pContribFromLibrary [15] *pSizeOfMovement)
-
(int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement))
-
(int) (Math.cos (72*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement));

yPContig=yCircle
    - (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (72*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (54*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
-
(int) (Math.sin (36*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))

+ (int) (Math.sin (18*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.sin (36*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
+ (int) (Math.sin (72*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
    + (int) (pContribFromLibrary [10] *pSizeOfMovement)

```

```

+ (int) (Math.sin(72*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement))
+ (int) (Math.sin(36*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
+ (int) (Math.sin(18*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))

-
(int) (Math.sin(18*Math.PI/180) * (pContribFromLibrary[16] * pSizeOfMovement))
-
(int) (Math.sin(36*Math.PI/180) * (pContribFromLibrary[17] * pSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (pContribFromLibrary[18] * pSizeOfMovement))
-
(int) (Math.sin(72*Math.PI/180) * (pContribFromLibrary[19] * pSizeOfMovement));

xEContig=xCircle
+ (int) (Math.cos(72*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(36*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (eContribFromLibrary[5] * eSizeOfMovement)
+ (int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.cos(36*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
+ (int) (Math.cos(72*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))

-
(int) (Math.cos(72*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
-
(int) (Math.cos(54*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
-
(int) (Math.cos(36*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
-
(int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
- (int) (eContribFromLibrary[15] * eSizeOfMovement)
-
(int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
-
(int) (Math.cos(36*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
-
(int) (Math.cos(54*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))
-
(int) (Math.cos(72*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement));

yEContig=yCircle
- (int) (eContribFromLibrary[0] * eSizeOfMovement)

```

```

-
(int) (Math.sin(72*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
-
(int) (Math.sin(36*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))

+ (int) (Math.sin(18*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.sin(36*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
+ (int) (Math.sin(72*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
    + (int) (eContribFromLibrary[10] * eSizeOfMovement)
+ (int) (Math.sin(72*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
+ (int) (Math.sin(36*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
+ (int) (Math.sin(18*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))

-
(int) (Math.sin(18*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
-
(int) (Math.sin(36*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))
-
(int) (Math.sin(72*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement));

```

xWContig=xCircle

```

+ (int) (Math.cos(72*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(36*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(18*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
    + (int) (wContribFromLibrary[5] * wSizeOfMovement)
+ (int) (Math.cos(18*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.cos(36*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))

    + (int) (Math.cos(72*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))

-
(int) (Math.cos(72*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))

```

```

-
(int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary[12] *wSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (wContribFromLibrary[13] *wSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary[14] *wSizeOfMovement))
- (int) (wContribFromLibrary[15] *wSizeOfMovement)
-
(int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary[16] *wSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (wContribFromLibrary[17] *wSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary[18] *wSizeOfMovement))
-
(int) (Math.cos (72*Math.PI/180) * (wContribFromLibrary[19] *wSizeOfMovement));

    yWContig=yCircle
        - (int) (wContribFromLibrary[0] *wSizeOfMovement)
        -
(int) (Math.sin (72*Math.PI/180) * (wContribFromLibrary[1] *wSizeOfMovement))
-
(int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary[2] *wSizeOfMovement))
-
(int) (Math.sin (36*Math.PI/180) * (wContribFromLibrary[3] *wSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary[4] *wSizeOfMovement))

+ (int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary[6] *wSizeOfMovement))
+ (int) (Math.sin (36*Math.PI/180) * (wContribFromLibrary[7] *wSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary[8] *wSizeOfMovement))
+ (int) (Math.sin (72*Math.PI/180) * (wContribFromLibrary[9] *wSizeOfMovement))
+ (int) (wContribFromLibrary[10] *wSizeOfMovement)
+ (int) (Math.sin (72*Math.PI/180) * (wContribFromLibrary[11] *wSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary[12] *wSizeOfMovement))
+ (int) (Math.sin (36*Math.PI/180) * (wContribFromLibrary[13] *wSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary[14] *wSizeOfMovement))

-
(int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary[16] *wSizeOfMovement))
-
(int) (Math.sin (36*Math.PI/180) * (wContribFromLibrary[17] *wSizeOfMovement))
-
(int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary[18] *wSizeOfMovement))
-
(int) (Math.sin (72*Math.PI/180) * (wContribFromLibrary[19] *wSizeOfMovement));
    } // end 20

    // if numberOfLibraries is 21, 17 plus one 20
    else if ( 21 == numberOfLibraries)

```

```

{
    xPContig=xCircle
+ (int) (Math.cos (73*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
+ (int) (Math.cos (56*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
+ (int) (Math.cos (39*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
+ (int) (Math.cos (22*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
+ (int) (Math.cos (5*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
+ (int) (Math.cos (12*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.cos (29*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.cos (46*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
        + (int) (Math.cos (63*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
        + (int) (Math.cos (80*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
(int) (Math.cos (83*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
-
(int) (Math.cos (49*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
-
(int) (Math.cos (32*Math.PI/180) * (pContribFromLibrary [14] *pSizeOfMovement))
-
(int) (Math.cos (15*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))
-
(int) (Math.cos (2*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))
-
(int) (Math.cos (19*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement))
-
(int) (Math.cos (53*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement))
-
(int) (Math.cos (70*Math.PI/180) * (pContribFromLibrary [20] *pSizeOfMovement));

    yPContig=yCircle
        - (int) (pContribFromLibrary [0] *pSizeOfMovement)'
-
(int) (Math.sin (73*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (56*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
-
(int) (Math.sin (39*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
-
(int) (Math.sin (22*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
-
(int) (Math.sin (5*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
+ (int) (Math.sin (12*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))

```

```

+ (int) (Math.sin(29*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
+ (int) (Math.sin(46*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))
+ (int) (Math.sin(63*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
+ (int) (Math.sin(80*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))
+ (int) (Math.sin(83*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
+ (int) (Math.sin(66*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement))
+ (int) (Math.sin(49*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
+ (int) (Math.sin(32*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))
+ (int) (Math.sin(15*Math.PI/180) * (pContribFromLibrary[15] * pSizeOfMovement))
-
(int) (Math.sin(2*Math.PI/180) * (pContribFromLibrary[16] * pSizeOfMovement))
-
(int) (Math.sin(19*Math.PI/180) * (pContribFromLibrary[17] * pSizeOfMovement))
-
(int) (Math.sin(36*Math.PI/180) * (pContribFromLibrary[18] * pSizeOfMovement))
-
(int) (Math.sin(53*Math.PI/180) * (pContribFromLibrary[19] * pSizeOfMovement))
-
(int) (Math.sin(70*Math.PI/180) * (pContribFromLibrary[20] * pSizeOfMovement));
    xEContig=xCircle

+ (int) (Math.cos(73*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(56*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(39*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(22*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.cos(5*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
+ (int) (Math.cos(12*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.cos(29*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.cos(46*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
    + (int) (Math.cos(63*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
    + (int) (Math.cos(80*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
-
(int) (Math.cos(83*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
-
(int) (Math.cos(49*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
-
(int) (Math.cos(32*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))

```

```

-
(int) (Math.cos (15*Math.PI/180) * (eContribFromLibrary [15] *eSizeOfMovement))
-
(int) (Math.cos (2*Math.PI/180) * (eContribFromLibrary [16] *eSizeOfMovement))
-
(int) (Math.cos (19*Math.PI/180) * (eContribFromLibrary [17] *eSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (eContribFromLibrary [18] *eSizeOfMovement))
-
(int) (Math.cos (53*Math.PI/180) * (eContribFromLibrary [19] *eSizeOfMovement))
-
(int) (Math.cos (70*Math.PI/180) * (eContribFromLibrary [20] *eSizeOfMovement));

```

```

yEContig=yCircle
- (int) (eContribFromLibrary [0] *eSizeOfMovement)
-
(int) (Math.sin (73*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
-
(int) (Math.sin (56*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
-
(int) (Math.sin (39*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
-
(int) (Math.sin (22*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
-
(int) (Math.sin (5*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))
+ (int) (Math.sin (12*Math.PI/180) * (eContribFromLibrary [6] *eSizeOfMovement))
+ (int) (Math.sin (29*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
+ (int) (Math.sin (46*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
+ (int) (Math.sin (63*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))
+ (int) (Math.sin (80*Math.PI/180) * (eContribFromLibrary [10] *eSizeOfMovement))
+ (int) (Math.sin (83*Math.PI/180) * (eContribFromLibrary [11] *eSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (eContribFromLibrary [12] *eSizeOfMovement))
+ (int) (Math.sin (49*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement))
+ (int) (Math.sin (32*Math.PI/180) * (eContribFromLibrary [14] *eSizeOfMovement))
+ (int) (Math.sin (15*Math.PI/180) * (eContribFromLibrary [15] *eSizeOfMovement))
-
(int) (Math.sin (2*Math.PI/180) * (eContribFromLibrary [16] *eSizeOfMovement))
-
(int) (Math.sin (19*Math.PI/180) * (eContribFromLibrary [17] *eSizeOfMovement))
-
(int) (Math.sin (36*Math.PI/180) * (eContribFromLibrary [18] *eSizeOfMovement))
-
(int) (Math.sin (53*Math.PI/180) * (eContribFromLibrary [19] *eSizeOfMovement))
-
(int) (Math.sin (70*Math.PI/180) * (eContribFromLibrary [20] *eSizeOfMovement));

```

```

xWContig=xCircle

```

```

+ (int) (Math.cos (73*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
+ (int) (Math.cos (56*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
+ (int) (Math.cos (39*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
+ (int) (Math.cos (22*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
+ (int) (Math.cos (5*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
+ (int) (Math.cos (12*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
+ (int) (Math.cos (29*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
+ (int) (Math.cos (46*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
+ (int) (Math.cos (63*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
+ (int) (Math.cos (80*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
-
(int) (Math.cos (83*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
-
(int) (Math.cos (49*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
-
(int) (Math.cos (32*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
-
(int) (Math.cos (15*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
-
(int) (Math.cos (2*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))
-
(int) (Math.cos (19*Math.PI/180) * (wContribFromLibrary [17] *wSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (wContribFromLibrary [18] *wSizeOfMovement))
-
(int) (Math.cos (53*Math.PI/180) * (wContribFromLibrary [19] *wSizeOfMovement))
-
(int) (Math.cos (70*Math.PI/180) * (wContribFromLibrary [20] *wSizeOfMovement));

yWContig=yCircle
- (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (73*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
-
(int) (Math.sin (56*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
-
(int) (Math.sin (39*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
-
(int) (Math.sin (22*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
-
(int) (Math.sin (5*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
+ (int) (Math.sin (12*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
+ (int) (Math.sin (29*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))

```

```

+ (int) (Math.sin(46*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
+ (int) (Math.sin(63*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
+ (int) (Math.sin(80*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
+ (int) (Math.sin(83*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
+ (int) (Math.sin(66*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
+ (int) (Math.sin(49*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
+ (int) (Math.sin(32*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
+ (int) (Math.sin(15*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))
-
(int) (Math.sin(2*Math.PI/180) * (wContribFromLibrary[16] * wSizeOfMovement))
-
(int) (Math.sin(19*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
-
(int) (Math.sin(36*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
-
(int) (Math.sin(53*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
-
(int) (Math.sin(70*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement));
} // end 21
} // end calcXY_16to21()
//*****
void calcXY_22to26()
//*****

{
    // if numberOfLibraries is 22, 16 plus eight 17-degree pies
    if ( 22 == numberOfLibraries)
    {
        xPContig=xCircle

+ (int) (Math.cos(74*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(58*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (Math.cos(26*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
+ (int) (Math.cos(10*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
+ (int) (Math.cos(22*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
+ (int) (Math.cos(38*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))

        + (int) (Math.cos(54*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
        + (int) (Math.cos(70*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))

```

```

+ (int) (Math.cos (86*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
-
(int) (Math.cos (62*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
-
(int) (Math.cos (46*Math.PI/180) * (pContribFromLibrary [14] *pSizeOfMovement))
-
(int) (Math.cos (29*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))
-
(int) (Math.cos (12*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))
-
(int) (Math.cos (5*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement))
-
(int) (Math.cos (22*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement))
-
(int) (Math.cos (39*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement))
-
(int) (Math.cos (56*Math.PI/180) * (pContribFromLibrary [20] *pSizeOfMovement))
-
(int) (Math.cos (73*Math.PI/180) * (pContribFromLibrary [21] *pSizeOfMovement));

    yPContig=yCircle
        - (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (74*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (58*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
-
(int) (Math.sin (42*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
-
(int) (Math.sin (26*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
-
(int) (Math.sin (10*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
+ (int) (Math.sin (6*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.sin (22*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.sin (38*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
+ (int) (Math.sin (70*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
+ (int) (Math.sin (86*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
+ (int) (Math.sin (62*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
+ (int) (Math.sin (46*Math.PI/180) * (pContribFromLibrary [14] *pSizeOfMovement))
+ (int) (Math.sin (29*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))
+ (int) (Math.sin (12*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))

```

```

-
(int) (Math.sin(5*Math.PI/180) * (pContribFromLibrary[17] * pSizeOfMovement))
-
(int) (Math.sin(22*Math.PI/180) * (pContribFromLibrary[18] * pSizeOfMovement))
-
(int) (Math.sin(39*Math.PI/180) * (pContribFromLibrary[19] * pSizeOfMovement))
-
(int) (Math.sin(56*Math.PI/180) * (pContribFromLibrary[20] * pSizeOfMovement))
-
(int) (Math.sin(73*Math.PI/180) * (pContribFromLibrary[21] * pSizeOfMovement));

    xEContig=xCircle

+ (int) (Math.cos(74*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(58*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(26*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.cos(10*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.cos(22*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.cos(38*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))

    + (int) (Math.cos(54*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
    + (int) (Math.cos(70*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))

+ (int) (Math.cos(86*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
-
(int) (Math.cos(78*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
-
(int) (Math.cos(62*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
-
(int) (Math.cos(46*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
-
(int) (Math.cos(29*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
-
(int) (Math.cos(12*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
-
(int) (Math.cos(5*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
-
(int) (Math.cos(22*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))
-
(int) (Math.cos(39*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
-
(int) (Math.cos(56*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
-
(int) (Math.cos(73*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement));

    yEContig=yCircle
        - (int) (eContribFromLibrary[0] * eSizeOfMovement)

```

```

-
(int) (Math.sin(74*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
-
(int) (Math.sin(58*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
-
(int) (Math.sin(26*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
-
(int) (Math.sin(10*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.sin(22*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.sin(38*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
+ (int) (Math.sin(70*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
+ (int) (Math.sin(86*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
+ (int) (Math.sin(62*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
+ (int) (Math.sin(46*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
+ (int) (Math.sin(29*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
+ (int) (Math.sin(12*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
-
(int) (Math.sin(5*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
-
(int) (Math.sin(22*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))
-
(int) (Math.sin(39*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
-
(int) (Math.sin(56*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
-
(int) (Math.sin(73*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement));

```

 xWContig=xCircle

```

+ (int) (Math.cos(74*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(58*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(26*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(10*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))

```

```

+ (int) (Math.cos (22*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
+ (int) (Math.cos (38*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
    + (int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
    + (int) (Math.cos (70*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
+ (int) (Math.cos (86*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
-
(int) (Math.cos (62*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
-
(int) (Math.cos (46*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
-
(int) (Math.cos (29*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
-
(int) (Math.cos (12*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))
-
(int) (Math.cos (5*Math.PI/180) * (wContribFromLibrary [17] *wSizeOfMovement))
-
(int) (Math.cos (22*Math.PI/180) * (wContribFromLibrary [18] *wSizeOfMovement))
-
(int) (Math.cos (39*Math.PI/180) * (wContribFromLibrary [19] *wSizeOfMovement))
-
(int) (Math.cos (56*Math.PI/180) * (wContribFromLibrary [20] *wSizeOfMovement))
-
(int) (Math.cos (73*Math.PI/180) * (wContribFromLibrary [21] *wSizeOfMovement));

yWContig=yCircle
    - (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (74*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
-
(int) (Math.sin (58*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
-
(int) (Math.sin (42*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
-
(int) (Math.sin (26*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
-
(int) (Math.sin (10*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
+ (int) (Math.sin (6*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
+ (int) (Math.sin (22*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
+ (int) (Math.sin (38*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
+ (int) (Math.sin (70*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
+ (int) (Math.sin (86*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))

```

```

+ (int) (Math.sin(62*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
+ (int) (Math.sin(46*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
+ (int) (Math.sin(29*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))
+ (int) (Math.sin(12*Math.PI/180) * (wContribFromLibrary[16] * wSizeOfMovement))
-
(int) (Math.sin(5*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
-
(int) (Math.sin(22*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
-
(int) (Math.sin(39*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
-
(int) (Math.sin(56*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))
-
(int) (Math.sin(73*Math.PI/180) * (wContribFromLibrary[21] * wSizeOfMovement));
} // end 22

// if numberOfLibraries is 23, 16 + eight 15-degree pies
else if ( 23 == numberOfLibraries)
{
    xPContig=xCircle

+ (int) (Math.cos(74*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(58*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (Math.cos(26*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
+ (int) (Math.cos(10*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
+ (int) (Math.cos(22*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
+ (int) (Math.cos(38*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))

    + (int) (Math.cos(54*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))

    + (int) (Math.cos(70*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))

+ (int) (Math.cos(86*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
-
(int) (Math.cos(78*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement))
-
(int) (Math.cos(62*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
-
(int) (Math.cos(46*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[15] * pSizeOfMovement))
-
(int) (Math.cos(15*Math.PI/180) * (pContribFromLibrary[16] * pSizeOfMovement))
- (int) (pContribFromLibrary[17] * pSizeOfMovement)

```

```

-
(int) (Math.cos (15*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement))
-
(int) (Math.cos (45*Math.PI/180) * (pContribFromLibrary [20] *pSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (pContribFromLibrary [21] *pSizeOfMovement))
-
(int) (Math.cos (75*Math.PI/180) * (pContribFromLibrary [22] *pSizeOfMovement));

    yPContig=yCircle
        - (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (74*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (58*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
-
(int) (Math.sin (42*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
-
(int) (Math.sin (26*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
-
(int) (Math.sin (10*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
+ (int) (Math.sin (6*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.sin (22*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.sin (38*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
+ (int) (Math.sin (70*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
+ (int) (Math.sin (86*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
+ (int) (Math.sin (62*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
+ (int) (Math.sin (46*Math.PI/180) * (pContribFromLibrary [14] *pSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))
+ (int) (Math.sin (15*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))

-
(int) (Math.sin (15*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement))
-
(int) (Math.sin (45*Math.PI/180) * (pContribFromLibrary [20] *pSizeOfMovement))
-
(int) (Math.sin (60*Math.PI/180) * (pContribFromLibrary [21] *pSizeOfMovement))
-
(int) (Math.sin (75*Math.PI/180) * (pContribFromLibrary [22] *pSizeOfMovement));

```

```

xEContig=xCircle
+ (int) (Math.cos (74*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
+ (int) (Math.cos (58*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
+ (int) (Math.cos (42*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
+ (int) (Math.cos (26*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
+ (int) (Math.cos (10*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))
+ (int) (Math.cos (6*Math.PI/180) * (eContribFromLibrary [6] *eSizeOfMovement))
+ (int) (Math.cos (22*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
+ (int) (Math.cos (38*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
+ (int) (Math.cos (54*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))
+ (int) (Math.cos (70*Math.PI/180) * (eContribFromLibrary [10] *eSizeOfMovement))
+ (int) (Math.cos (86*Math.PI/180) * (eContribFromLibrary [11] *eSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (eContribFromLibrary [12] *eSizeOfMovement))
-
(int) (Math.cos (62*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement))
-
(int) (Math.cos (46*Math.PI/180) * (eContribFromLibrary [14] *eSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [15] *eSizeOfMovement))
-
(int) (Math.cos (15*Math.PI/180) * (eContribFromLibrary [16] *eSizeOfMovement))
- (int) (eContribFromLibrary [17] *eSizeOfMovement)
-
(int) (Math.cos (15*Math.PI/180) * (eContribFromLibrary [18] *eSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [19] *eSizeOfMovement))
-
(int) (Math.cos (45*Math.PI/180) * (eContribFromLibrary [20] *eSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (eContribFromLibrary [21] *eSizeOfMovement))
-
(int) (Math.cos (75*Math.PI/180) * (eContribFromLibrary [22] *eSizeOfMovement));

yEContig=yCircle
- (int) (eContribFromLibrary [0] *eSizeOfMovement)
-
(int) (Math.sin (74*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
-
(int) (Math.sin (58*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
-
(int) (Math.sin (42*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
-
(int) (Math.sin (26*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
-
(int) (Math.sin (10*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))

```

```

+ (int) (Math.sin(6*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.sin(22*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.sin(38*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
+ (int) (Math.sin(70*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
+ (int) (Math.sin(86*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
+ (int) (Math.sin(62*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
+ (int) (Math.sin(46*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
+ (int) (Math.sin(15*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))

-
(int) (Math.sin(15*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
-
(int) (Math.sin(45*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
-
(int) (Math.sin(60*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement))
-
(int) (Math.sin(75*Math.PI/180) * (eContribFromLibrary[22] * eSizeOfMovement));
    xWContig=xCircle

+ (int) (Math.cos(74*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(58*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(26*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(10*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.cos(22*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.cos(38*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
    + (int) (Math.cos(54*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
    + (int) (Math.cos(70*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
+ (int) (Math.cos(86*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))

```

```

-
(int) (Math.cos (78*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
-
(int) (Math.cos (62*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
-
(int) (Math.cos (46*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
-
(int) (Math.cos (15*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))
- (int) (wContribFromLibrary [17] *wSizeOfMovement)
-
(int) (Math.cos (15*Math.PI/180) * (wContribFromLibrary [18] *wSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (wContribFromLibrary [19] *wSizeOfMovement))
-
(int) (Math.cos (45*Math.PI/180) * (wContribFromLibrary [20] *wSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (wContribFromLibrary [21] *wSizeOfMovement))
-
(int) (Math.cos (75*Math.PI/180) * (wContribFromLibrary [22] *wSizeOfMovement));

    yWContig=yCircle
        - (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (74*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
-
(int) (Math.sin (58*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
-
(int) (Math.sin (42*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
-
(int) (Math.sin (26*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
-
(int) (Math.sin (10*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
+ (int) (Math.sin (6*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
+ (int) (Math.sin (22*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
+ (int) (Math.sin (38*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
+ (int) (Math.sin (70*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
+ (int) (Math.sin (86*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
+ (int) (Math.sin (62*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
+ (int) (Math.sin (46*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
+ (int) (Math.sin (15*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))

```

```

-
(int) (Math.sin(15*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
-
(int) (Math.sin(45*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))
-
(int) (Math.sin(60*Math.PI/180) * (wContribFromLibrary[21] * wSizeOfMovement))
-
(int) (Math.sin(75*Math.PI/180) * (wContribFromLibrary[22] * wSizeOfMovement));
    } // end 23

    // if numberOfLibraries is 24, all 15-degree
    else if ( 24 == numberOfLibraries)
    {
        xPContig=xCircle

+ (int) (Math.cos(75*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(60*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
+ (int) (Math.cos(15*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
    + (int) (pContribFromLibrary[6] * pSizeOfMovement)
+ (int) (Math.cos(15*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))
    + (int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
    + (int) (Math.cos(60*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))
+ (int) (Math.cos(75*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))

-
(int) (Math.cos(75*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
-
(int) (Math.cos(60*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))
-
(int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[15] * pSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[16] * pSizeOfMovement))
-
(int) (Math.cos(15*Math.PI/180) * (pContribFromLibrary[17] * pSizeOfMovement))
    - (int) (pContribFromLibrary[18] * pSizeOfMovement)
-
(int) (Math.cos(15*Math.PI/180) * (pContribFromLibrary[19] * pSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[20] * pSizeOfMovement))
-
(int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[21] * pSizeOfMovement))
-
(int) (Math.cos(60*Math.PI/180) * (pContribFromLibrary[22] * pSizeOfMovement))

```

```

-
(int) (Math.cos (75*Math.PI/180) * (pContribFromLibrary [23] *pSizeOfMovement)) ;

    yPContig=yCircle
        - (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (75*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (60*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
-
(int) (Math.sin (45*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
-
(int) (Math.sin (15*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))

+ (int) (Math.sin (15*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
+ (int) (Math.sin (45*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
+ (int) (Math.sin (60*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
+ (int) (Math.sin (75*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
    + (int) (pContribFromLibrary [12] *pSizeOfMovement)

+ (int) (Math.sin (75*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
+ (int) (Math.sin (60*Math.PI/180) * (pContribFromLibrary [14] *pSizeOfMovement))
+ (int) (Math.sin (45*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))
+ (int) (Math.sin (15*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement))

-
(int) (Math.sin (15*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [20] *pSizeOfMovement))
-
(int) (Math.sin (45*Math.PI/180) * (pContribFromLibrary [21] *pSizeOfMovement))
-
(int) (Math.sin (60*Math.PI/180) * (pContribFromLibrary [22] *pSizeOfMovement))
-
(int) (Math.sin (75*Math.PI/180) * (pContribFromLibrary [23] *pSizeOfMovement)) ;

    xEContig=xCircle

+ (int) (Math.cos (75*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
+ (int) (Math.cos (60*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
+ (int) (Math.cos (45*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))

```

```

+(int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
+(int) (Math.cos (15*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))
      +(int) (eContribFromLibrary [6] *eSizeOfMovement)
+(int) (Math.cos (15*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
+(int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
      +(int) (Math.cos (45*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))
      +(int) (Math.cos (60*Math.PI/180) * (eContribFromLibrary [10] *eSizeOfMovement))
+(int) (Math.cos (75*Math.PI/180) * (eContribFromLibrary [11] *eSizeOfMovement))

-
(int) (Math.cos (75*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (eContribFromLibrary [14] *eSizeOfMovement))
-
(int) (Math.cos (45*Math.PI/180) * (eContribFromLibrary [15] *eSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [16] *eSizeOfMovement))
-
(int) (Math.cos (15*Math.PI/180) * (eContribFromLibrary [17] *eSizeOfMovement))
      - (int) (eContribFromLibrary [18] *eSizeOfMovement)
-
(int) (Math.cos (15*Math.PI/180) * (eContribFromLibrary [19] *eSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [20] *eSizeOfMovement))
-
(int) (Math.cos (45*Math.PI/180) * (eContribFromLibrary [21] *eSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (eContribFromLibrary [22] *eSizeOfMovement))
-
(int) (Math.cos (75*Math.PI/180) * (eContribFromLibrary [23] *eSizeOfMovement));

yEContig=yCircle
      - (int) (eContribFromLibrary [0] *eSizeOfMovement)
-
(int) (Math.sin (75*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
-
(int) (Math.sin (60*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
-
(int) (Math.sin (45*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
-
(int) (Math.sin (15*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))

+(int) (Math.sin (15*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
+(int) (Math.sin (30*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
+(int) (Math.sin (45*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))

```

```

+ (int) (Math.sin(60*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
+ (int) (Math.sin(75*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
    + (int) (eContribFromLibrary[12] * eSizeOfMovement)
+ (int) (Math.sin(75*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
+ (int) (Math.sin(60*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
+ (int) (Math.sin(45*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
+ (int) (Math.sin(15*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))

```

```

-
(int) (Math.sin(15*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
-
(int) (Math.sin(45*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement))
-
(int) (Math.sin(60*Math.PI/180) * (eContribFromLibrary[22] * eSizeOfMovement))
-
(int) (Math.sin(75*Math.PI/180) * (eContribFromLibrary[23] * eSizeOfMovement));

```

 xWContig=xCircle

```

+ (int) (Math.cos(75*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(60*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(45*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(15*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
    + (int) (wContribFromLibrary[6] * wSizeOfMovement)
+ (int) (Math.cos(15*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
    + (int) (Math.cos(45*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
    + (int) (Math.cos(60*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
+ (int) (Math.cos(75*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
-
(int) (Math.cos(75*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
-
(int) (Math.cos(60*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
-
(int) (Math.cos(45*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))

```

```

-
(int) (Math.cos (30*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))
-
(int) (Math.cos (15*Math.PI/180) * (wContribFromLibrary [17] *wSizeOfMovement))
- (int) (wContribFromLibrary [18] *wSizeOfMovement)
-
(int) (Math.cos (15*Math.PI/180) * (wContribFromLibrary [19] *wSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (wContribFromLibrary [20] *wSizeOfMovement))
-
(int) (Math.cos (45*Math.PI/180) * (wContribFromLibrary [21] *wSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (wContribFromLibrary [22] *wSizeOfMovement))
-
(int) (Math.cos (75*Math.PI/180) * (wContribFromLibrary [23] *wSizeOfMovement));

yWContig=yCircle
- (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (75*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
-
(int) (Math.sin (60*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
-
(int) (Math.sin (45*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
-
(int) (Math.sin (15*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))

+ (int) (Math.sin (15*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
+ (int) (Math.sin (45*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
+ (int) (Math.sin (60*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
+ (int) (Math.sin (75*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
+ (int) (wContribFromLibrary [12] *wSizeOfMovement)
+ (int) (Math.sin (75*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
+ (int) (Math.sin (60*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
+ (int) (Math.sin (45*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))
+ (int) (Math.sin (15*Math.PI/180) * (wContribFromLibrary [17] *wSizeOfMovement))

-
(int) (Math.sin (15*Math.PI/180) * (wContribFromLibrary [19] *wSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary [20] *wSizeOfMovement))
-
(int) (Math.sin (45*Math.PI/180) * (wContribFromLibrary [21] *wSizeOfMovement))

```

```

-
(int) (Math.sin(60*Math.PI/180) * (wContribFromLibrary[22] * wSizeOfMovement))
-
(int) (Math.sin(75*Math.PI/180) * (wContribFromLibrary[23] * wSizeOfMovement));
} // end 24

// if numberOfLibraries is 25, fifteen 14-degree & ten 15-degree pies
else if ( 25 == numberOfLibraries)
{
    xPContig=xCircle

+ (int) (Math.cos(75*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(60*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
+ (int) (Math.cos(15*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
    + (int) (pContribFromLibrary[6] * pSizeOfMovement)
+ (int) (Math.cos(15*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))
    + (int) (Math.cos(45*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
    + (int) (Math.cos(60*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))
+ (int) (Math.cos(74*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
+ (int) (Math.cos(88*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement))
-
(int) (Math.cos(78*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
-
(int) (Math.cos(64*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))
-
(int) (Math.cos(50*Math.PI/180) * (pContribFromLibrary[15] * pSizeOfMovement))
-
(int) (Math.cos(36*Math.PI/180) * (pContribFromLibrary[16] * pSizeOfMovement))
-
(int) (Math.cos(22*Math.PI/180) * (pContribFromLibrary[17] * pSizeOfMovement))
-
(int) (Math.cos(8*Math.PI/180) * (pContribFromLibrary[18] * pSizeOfMovement))
-
(int) (Math.cos(6*Math.PI/180) * (pContribFromLibrary[19] * pSizeOfMovement))
-
(int) (Math.cos(20*Math.PI/180) * (pContribFromLibrary[20] * pSizeOfMovement))
-
(int) (Math.cos(34*Math.PI/180) * (pContribFromLibrary[21] * pSizeOfMovement))
-
(int) (Math.cos(48*Math.PI/180) * (pContribFromLibrary[22] * pSizeOfMovement))
-
(int) (Math.cos(62*Math.PI/180) * (pContribFromLibrary[23] * pSizeOfMovement))
-
(int) (Math.cos(76*Math.PI/180) * (pContribFromLibrary[24] * pSizeOfMovement));

```

```

yPContig=yCircle
- (int) (pContribFromLibrary[0] *pSizeOfMovement)
-
(int) (Math.sin(75*Math.PI/180) * (pContribFromLibrary[1] *pSizeOfMovement))
-
(int) (Math.sin(60*Math.PI/180) * (pContribFromLibrary[2] *pSizeOfMovement))
-
(int) (Math.sin(45*Math.PI/180) * (pContribFromLibrary[3] *pSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[4] *pSizeOfMovement))
-
(int) (Math.sin(15*Math.PI/180) * (pContribFromLibrary[5] *pSizeOfMovement))

+ (int) (Math.sin(15*Math.PI/180) * (pContribFromLibrary[7] *pSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[8] *pSizeOfMovement))
+ (int) (Math.sin(45*Math.PI/180) * (pContribFromLibrary[9] *pSizeOfMovement))
+ (int) (Math.sin(60*Math.PI/180) * (pContribFromLibrary[10] *pSizeOfMovement))
+ (int) (Math.sin(74*Math.PI/180) * (pContribFromLibrary[11] *pSizeOfMovement))
+ (int) (Math.sin(88*Math.PI/180) * (pContribFromLibrary[12] *pSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (pContribFromLibrary[13] *pSizeOfMovement))
+ (int) (Math.sin(64*Math.PI/180) * (pContribFromLibrary[14] *pSizeOfMovement))
+ (int) (Math.sin(50*Math.PI/180) * (pContribFromLibrary[15] *pSizeOfMovement))
+ (int) (Math.sin(36*Math.PI/180) * (pContribFromLibrary[16] *pSizeOfMovement))
+ (int) (Math.sin(22*Math.PI/180) * (pContribFromLibrary[17] *pSizeOfMovement))
+ (int) (Math.sin(8*Math.PI/180) * (pContribFromLibrary[18] *pSizeOfMovement))
-
(int) (Math.sin(6*Math.PI/180) * (pContribFromLibrary[19] *pSizeOfMovement))
-
(int) (Math.sin(20*Math.PI/180) * (pContribFromLibrary[20] *pSizeOfMovement))
-
(int) (Math.sin(34*Math.PI/180) * (pContribFromLibrary[21] *pSizeOfMovement))
-
(int) (Math.sin(48*Math.PI/180) * (pContribFromLibrary[22] *pSizeOfMovement))
-
(int) (Math.sin(62*Math.PI/180) * (pContribFromLibrary[23] *pSizeOfMovement))
-
(int) (Math.sin(76*Math.PI/180) * (pContribFromLibrary[24] *pSizeOfMovement));

xEContig=xCircle

+ (int) (Math.cos(75*Math.PI/180) * (eContribFromLibrary[1] *eSizeOfMovement))
+ (int) (Math.cos(60*Math.PI/180) * (eContribFromLibrary[2] *eSizeOfMovement))

```

```

+ (int) (Math.cos (45*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
+ (int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
+ (int) (Math.cos (15*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))
    + (int) (eContribFromLibrary [6] *eSizeOfMovement)
+ (int) (Math.cos (15*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
+ (int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
    + (int) (Math.cos (45*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))
    + (int) (Math.cos (60*Math.PI/180) * (eContribFromLibrary [10] *eSizeOfMovement))
+ (int) (Math.cos (74*Math.PI/180) * (eContribFromLibrary [11] *eSizeOfMovement))
+ (int) (Math.cos (88*Math.PI/180) * (eContribFromLibrary [12] *eSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement))
-
(int) (Math.cos (64*Math.PI/180) * (eContribFromLibrary [14] *eSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (eContribFromLibrary [15] *eSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (eContribFromLibrary [16] *eSizeOfMovement))
-
(int) (Math.cos (22*Math.PI/180) * (eContribFromLibrary [17] *eSizeOfMovement))
-
(int) (Math.cos (8*Math.PI/180) * (eContribFromLibrary [18] *eSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (eContribFromLibrary [19] *eSizeOfMovement))
-
(int) (Math.cos (20*Math.PI/180) * (eContribFromLibrary [20] *eSizeOfMovement))
-
(int) (Math.cos (34*Math.PI/180) * (eContribFromLibrary [21] *eSizeOfMovement))
-
(int) (Math.cos (48*Math.PI/180) * (eContribFromLibrary [22] *eSizeOfMovement))
-
(int) (Math.cos (62*Math.PI/180) * (eContribFromLibrary [23] *eSizeOfMovement))
-
(int) (Math.cos (76*Math.PI/180) * (eContribFromLibrary [24] *eSizeOfMovement));

yEContig=yCircle
    - (int) (eContribFromLibrary [0] *eSizeOfMovement)
-
(int) (Math.sin (75*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
-
(int) (Math.sin (60*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
-
(int) (Math.sin (45*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
-
(int) (Math.sin (15*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))

```

```

+ (int) (Math.sin(15*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
+ (int) (Math.sin(45*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
+ (int) (Math.sin(60*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
+ (int) (Math.sin(74*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.sin(88*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
+ (int) (Math.sin(64*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
+ (int) (Math.sin(50*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
+ (int) (Math.sin(36*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
+ (int) (Math.sin(22*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
+ (int) (Math.sin(8*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))
-
(int) (Math.sin(6*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
-
(int) (Math.sin(20*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
-
(int) (Math.sin(34*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement))
-
(int) (Math.sin(48*Math.PI/180) * (eContribFromLibrary[22] * eSizeOfMovement))
-
(int) (Math.sin(62*Math.PI/180) * (eContribFromLibrary[23] * eSizeOfMovement))
-
(int) (Math.sin(76*Math.PI/180) * (eContribFromLibrary[24] * eSizeOfMovement));

    xWContig=xCircle

+ (int) (Math.cos(75*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(60*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(45*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(15*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
    + (int) (wContribFromLibrary[6] * wSizeOfMovement)
+ (int) (Math.cos(15*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))

    + (int) (Math.cos(45*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
    + (int) (Math.cos(60*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))

```

```

+ (int) (Math.cos (74*Math.PI/180) * (wContribFromLibrary[11]*wSizeOfMovement))
+ (int) (Math.cos (88*Math.PI/180) * (wContribFromLibrary[12]*wSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (wContribFromLibrary[13]*wSizeOfMovement))
-
(int) (Math.cos (64*Math.PI/180) * (wContribFromLibrary[14]*wSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (wContribFromLibrary[15]*wSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (wContribFromLibrary[16]*wSizeOfMovement))
-
(int) (Math.cos (22*Math.PI/180) * (wContribFromLibrary[17]*wSizeOfMovement))
-
(int) (Math.cos (8*Math.PI/180) * (wContribFromLibrary[18]*wSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (wContribFromLibrary[19]*wSizeOfMovement))
-
(int) (Math.cos (20*Math.PI/180) * (wContribFromLibrary[20]*wSizeOfMovement))
-
(int) (Math.cos (34*Math.PI/180) * (wContribFromLibrary[21]*wSizeOfMovement))
-
(int) (Math.cos (48*Math.PI/180) * (wContribFromLibrary[22]*wSizeOfMovement))
-
(int) (Math.cos (62*Math.PI/180) * (wContribFromLibrary[23]*wSizeOfMovement))
-
(int) (Math.cos (76*Math.PI/180) * (wContribFromLibrary[24]*wSizeOfMovement));

    yWContig=yCircle
        - (int) (wContribFromLibrary[0]*wSizeOfMovement)
        -
(int) (Math.sin (75*Math.PI/180) * (wContribFromLibrary[1]*wSizeOfMovement))
        -
(int) (Math.sin (60*Math.PI/180) * (wContribFromLibrary[2]*wSizeOfMovement))
        -
(int) (Math.sin (45*Math.PI/180) * (wContribFromLibrary[3]*wSizeOfMovement))
        -
(int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary[4]*wSizeOfMovement))
        -
(int) (Math.sin (15*Math.PI/180) * (wContribFromLibrary[5]*wSizeOfMovement))

+ (int) (Math.sin (15*Math.PI/180) * (wContribFromLibrary[7]*wSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary[8]*wSizeOfMovement))
+ (int) (Math.sin (45*Math.PI/180) * (wContribFromLibrary[9]*wSizeOfMovement))
+ (int) (Math.sin (60*Math.PI/180) * (wContribFromLibrary[10]*wSizeOfMovement))
+ (int) (Math.sin (74*Math.PI/180) * (wContribFromLibrary[11]*wSizeOfMovement))
+ (int) (Math.sin (88*Math.PI/180) * (wContribFromLibrary[12]*wSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (wContribFromLibrary[13]*wSizeOfMovement))

```

```

+ (int) (Math.sin(64*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
+ (int) (Math.sin(50*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))
+ (int) (Math.sin(36*Math.PI/180) * (wContribFromLibrary[16] * wSizeOfMovement))
+ (int) (Math.sin(22*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
+ (int) (Math.sin(8*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
-
(int) (Math.sin(6*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
-
(int) (Math.sin(20*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))
-
(int) (Math.sin(34*Math.PI/180) * (wContribFromLibrary[21] * wSizeOfMovement))
-
(int) (Math.sin(48*Math.PI/180) * (wContribFromLibrary[22] * wSizeOfMovement))
-
(int) (Math.sin(62*Math.PI/180) * (wContribFromLibrary[23] * wSizeOfMovement))
-
(int) (Math.sin(76*Math.PI/180) * (wContribFromLibrary[24] * wSizeOfMovement));
} // end 25

// if numberOfLibraries is 26
else if ( 26 == numberOfLibraries)
{
    xPContig=xCircle
+ (int) (Math.cos(76*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(62*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.cos(48*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (Math.cos(34*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
+ (int) (Math.cos(20*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
+ (int) (Math.cos(8*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
+ (int) (Math.cos(22*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))
+ (int) (Math.cos(36*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
+ (int) (Math.cos(50*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))
+ (int) (Math.cos(64*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement))
-
(int) (Math.cos(88*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
-
(int) (Math.cos(74*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))

```

```

-
(int) (Math.cos (60*Math.PI/180) * (pContribFromLibrary[15] *pSizeOfMovement))
-
(int) (Math.cos (46*Math.PI/180) * (pContribFromLibrary[16] *pSizeOfMovement))
-
(int) (Math.cos (32*Math.PI/180) * (pContribFromLibrary[17] *pSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary[18] *pSizeOfMovement))
-
(int) (Math.cos (4*Math.PI/180) * (pContribFromLibrary[19] *pSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (pContribFromLibrary[20] *pSizeOfMovement))
-
(int) (Math.cos (24*Math.PI/180) * (pContribFromLibrary[21] *pSizeOfMovement))
-
(int) (Math.cos (38*Math.PI/180) * (pContribFromLibrary[22] *pSizeOfMovement))
-
(int) (Math.cos (51*Math.PI/180) * (pContribFromLibrary[23] *pSizeOfMovement))
-
(int) (Math.cos (64*Math.PI/180) * (pContribFromLibrary[24] *pSizeOfMovement))
-
(int) (Math.cos (77*Math.PI/180) * (pContribFromLibrary[25] *pSizeOfMovement));

    yPContig=yCircle
        - (int) (pContribFromLibrary[0] *pSizeOfMovement)
-
(int) (Math.sin (76*Math.PI/180) * (pContribFromLibrary[1] *pSizeOfMovement))
-
(int) (Math.sin (62*Math.PI/180) * (pContribFromLibrary[2] *pSizeOfMovement))
-
(int) (Math.sin (48*Math.PI/180) * (pContribFromLibrary[3] *pSizeOfMovement))
-
(int) (Math.sin (34*Math.PI/180) * (pContribFromLibrary[4] *pSizeOfMovement))
-
(int) (Math.sin (20*Math.PI/180) * (pContribFromLibrary[5] *pSizeOfMovement))
-
(int) (Math.sin (6*Math.PI/180) * (pContribFromLibrary[6] *pSizeOfMovement))
+ (int) (Math.sin (8*Math.PI/180) * (pContribFromLibrary[7] *pSizeOfMovement))
+ (int) (Math.sin (22*Math.PI/180) * (pContribFromLibrary[8] *pSizeOfMovement))
+ (int) (Math.sin (36*Math.PI/180) * (pContribFromLibrary[9] *pSizeOfMovement))
+ (int) (Math.sin (50*Math.PI/180) * (pContribFromLibrary[10] *pSizeOfMovement))
+ (int) (Math.sin (64*Math.PI/180) * (pContribFromLibrary[11] *pSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (pContribFromLibrary[12] *pSizeOfMovement))
+ (int) (Math.sin (88*Math.PI/180) * (pContribFromLibrary[13] *pSizeOfMovement))
+ (int) (Math.sin (74*Math.PI/180) * (pContribFromLibrary[14] *pSizeOfMovement))
+ (int) (Math.sin (60*Math.PI/180) * (pContribFromLibrary[15] *pSizeOfMovement))
+ (int) (Math.sin (46*Math.PI/180) * (pContribFromLibrary[16] *pSizeOfMovement))

```

```

+ (int) (Math.sin(32*Math.PI/180) * (pContribFromLibrary[17] * pSizeOfMovement))
+ (int) (Math.sin(18*Math.PI/180) * (pContribFromLibrary[18] * pSizeOfMovement))
+ (int) (Math.sin(4*Math.PI/180) * (pContribFromLibrary[19] * pSizeOfMovement))
-
(int) (Math.sin(10*Math.PI/180) * (pContribFromLibrary[20] * pSizeOfMovement))
-
(int) (Math.sin(24*Math.PI/180) * (pContribFromLibrary[21] * pSizeOfMovement))
-
(int) (Math.sin(38*Math.PI/180) * (pContribFromLibrary[22] * pSizeOfMovement))
-
(int) (Math.sin(51*Math.PI/180) * (pContribFromLibrary[23] * pSizeOfMovement))
-
(int) (Math.sin(64*Math.PI/180) * (pContribFromLibrary[24] * pSizeOfMovement))
-
(int) (Math.sin(77*Math.PI/180) * (pContribFromLibrary[25] * pSizeOfMovement));

    xEContig=xCircle

+ (int) (Math.cos(76*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(62*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(48*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(34*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.cos(20*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
    + (int) (Math.cos(6*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.cos(8*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.cos(22*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
    + (int) (Math.cos(36*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
    + (int) (Math.cos(50*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
+ (int) (Math.cos(64*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
-
(int) (Math.cos(88*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
-
(int) (Math.cos(74*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
-
(int) (Math.cos(60*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
-
(int) (Math.cos(46*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
-
(int) (Math.cos(32*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
-
(int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))

```

```

-
(int) (Math.cos (4*Math.PI/180) * (eContribFromLibrary [19] *eSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (eContribFromLibrary [20] *eSizeOfMovement))
-
(int) (Math.cos (24*Math.PI/180) * (eContribFromLibrary [21] *eSizeOfMovement))
-
(int) (Math.cos (38*Math.PI/180) * (eContribFromLibrary [22] *eSizeOfMovement))
-
(int) (Math.cos (51*Math.PI/180) * (eContribFromLibrary [23] *eSizeOfMovement))
-
(int) (Math.cos (64*Math.PI/180) * (eContribFromLibrary [24] *eSizeOfMovement))
-
(int) (Math.cos (77*Math.PI/180) * (eContribFromLibrary [25] *eSizeOfMovement));

yEContig=yCircle
- (int) (eContribFromLibrary [0] *eSizeOfMovement)
-
(int) (Math.sin (76*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
-
(int) (Math.sin (62*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
-
(int) (Math.sin (48*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
-
(int) (Math.sin (34*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
-
(int) (Math.sin (20*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))
-
(int) (Math.sin (6*Math.PI/180) * (eContribFromLibrary [6] *eSizeOfMovement))
+ (int) (Math.sin (8*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
+ (int) (Math.sin (22*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
+ (int) (Math.sin (36*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))
+ (int) (Math.sin (50*Math.PI/180) * (eContribFromLibrary [10] *eSizeOfMovement))
+ (int) (Math.sin (64*Math.PI/180) * (eContribFromLibrary [11] *eSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (eContribFromLibrary [12] *eSizeOfMovement))
+ (int) (Math.sin (88*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement))
+ (int) (Math.sin (74*Math.PI/180) * (eContribFromLibrary [14] *eSizeOfMovement))
+ (int) (Math.sin (60*Math.PI/180) * (eContribFromLibrary [15] *eSizeOfMovement))
+ (int) (Math.sin (46*Math.PI/180) * (eContribFromLibrary [16] *eSizeOfMovement))
+ (int) (Math.sin (32*Math.PI/180) * (eContribFromLibrary [17] *eSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (eContribFromLibrary [18] *eSizeOfMovement))
+ (int) (Math.sin (4*Math.PI/180) * (eContribFromLibrary [19] *eSizeOfMovement))
-
(int) (Math.sin (10*Math.PI/180) * (eContribFromLibrary [20] *eSizeOfMovement))

```

```

-
(int) (Math.sin (24*Math.PI/180) * (eContribFromLibrary [21] *eSizeOfMovement))
-
(int) (Math.sin (38*Math.PI/180) * (eContribFromLibrary [22] *eSizeOfMovement))
-
(int) (Math.sin (51*Math.PI/180) * (eContribFromLibrary [23] *eSizeOfMovement))
-
(int) (Math.sin (64*Math.PI/180) * (eContribFromLibrary [24] *eSizeOfMovement))
-
(int) (Math.sin (77*Math.PI/180) * (eContribFromLibrary [25] *eSizeOfMovement));

    xWContig=xCircle

+ (int) (Math.cos (76*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
+ (int) (Math.cos (62*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
+ (int) (Math.cos (48*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
+ (int) (Math.cos (34*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
+ (int) (Math.cos (20*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))

    + (int) (Math.cos (6*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
+ (int) (Math.cos (8*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
+ (int) (Math.cos (22*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))

    + (int) (Math.cos (36*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
    + (int) (Math.cos (50*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
+ (int) (Math.cos (64*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
+ (int) (Math.cos (78*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
-
(int) (Math.cos (88*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
-
(int) (Math.cos (74*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
-
(int) (Math.cos (46*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))
-
(int) (Math.cos (32*Math.PI/180) * (wContribFromLibrary [17] *wSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary [18] *wSizeOfMovement))
-
(int) (Math.cos (4*Math.PI/180) * (wContribFromLibrary [19] *wSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (wContribFromLibrary [20] *wSizeOfMovement))
-
(int) (Math.cos (24*Math.PI/180) * (wContribFromLibrary [21] *wSizeOfMovement))
-
(int) (Math.cos (38*Math.PI/180) * (wContribFromLibrary [22] *wSizeOfMovement))

```

```

-
(int) (Math.cos (51*Math.PI/180) * (wContribFromLibrary [23] *wSizeOfMovement))
-
(int) (Math.cos (64*Math.PI/180) * (wContribFromLibrary [24] *wSizeOfMovement))
-
(int) (Math.cos (77*Math.PI/180) * (wContribFromLibrary [25] *wSizeOfMovement));

    yWContig=yCircle
        - (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (76*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
-
(int) (Math.sin (62*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
-
(int) (Math.sin (48*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
-
(int) (Math.sin (34*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
-
(int) (Math.sin (20*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
-
(int) (Math.sin (6*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
+ (int) (Math.sin (8*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
+ (int) (Math.sin (22*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
+ (int) (Math.sin (36*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
+ (int) (Math.sin (50*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
+ (int) (Math.sin (64*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
+ (int) (Math.sin (88*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
+ (int) (Math.sin (74*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
+ (int) (Math.sin (60*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
+ (int) (Math.sin (46*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))
+ (int) (Math.sin (32*Math.PI/180) * (wContribFromLibrary [17] *wSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary [18] *wSizeOfMovement))
+ (int) (Math.sin (4*Math.PI/180) * (wContribFromLibrary [19] *wSizeOfMovement))
-
(int) (Math.sin (10*Math.PI/180) * (wContribFromLibrary [20] *wSizeOfMovement))
-
(int) (Math.sin (24*Math.PI/180) * (wContribFromLibrary [21] *wSizeOfMovement))
-
(int) (Math.sin (38*Math.PI/180) * (wContribFromLibrary [22] *wSizeOfMovement))
-
(int) (Math.sin (51*Math.PI/180) * (wContribFromLibrary [23] *wSizeOfMovement))
-
(int) (Math.sin (64*Math.PI/180) * (wContribFromLibrary [24] *wSizeOfMovement))

```

```

(int) (Math.sin(77*Math.PI/180)*(wContribFromLibrary[25]*wSizeOfMovement));
    } // end 26
} // end calcXY_22to26()
//*****
void calcXY_27to30()
//*****

{
    // if numberOfLibraries is 27, eighteen 13- & nine 14-degree pies
    if ( 27 == numberOfLibraries)
    {
        xPContig=xCircle

+ (int) (Math.cos(76*Math.PI/180)*(pContribFromLibrary[1]*pSizeOfMovement))
+ (int) (Math.cos(62*Math.PI/180)*(pContribFromLibrary[2]*pSizeOfMovement))
+ (int) (Math.cos(48*Math.PI/180)*(pContribFromLibrary[3]*pSizeOfMovement))
+ (int) (Math.cos(34*Math.PI/180)*(pContribFromLibrary[4]*pSizeOfMovement))
+ (int) (Math.cos(20*Math.PI/180)*(pContribFromLibrary[5]*pSizeOfMovement))

        + (int) (Math.cos(6*Math.PI/180)*(pContribFromLibrary[6]*pSizeOfMovement))
+ (int) (Math.cos(8*Math.PI/180)*(pContribFromLibrary[7]*pSizeOfMovement))
+ (int) (Math.cos(22*Math.PI/180)*(pContribFromLibrary[8]*pSizeOfMovement))

        + (int) (Math.cos(36*Math.PI/180)*(pContribFromLibrary[9]*pSizeOfMovement))
        + (int) (Math.cos(49*Math.PI/180)*(pContribFromLibrary[10]*pSizeOfMovement))
+ (int) (Math.cos(62*Math.PI/180)*(pContribFromLibrary[11]*pSizeOfMovement))
+ (int) (Math.cos(75*Math.PI/180)*(pContribFromLibrary[12]*pSizeOfMovement))
+ (int) (Math.cos(88*Math.PI/180)*(pContribFromLibrary[13]*pSizeOfMovement))
(int) (Math.cos(79*Math.PI/180)*(pContribFromLibrary[14]*pSizeOfMovement))
(int) (Math.cos(66*Math.PI/180)*(pContribFromLibrary[15]*pSizeOfMovement))
(int) (Math.cos(53*Math.PI/180)*(pContribFromLibrary[16]*pSizeOfMovement))
(int) (Math.cos(40*Math.PI/180)*(pContribFromLibrary[17]*pSizeOfMovement))
(int) (Math.cos(27*Math.PI/180)*(pContribFromLibrary[18]*pSizeOfMovement))
(int) (Math.cos(14*Math.PI/180)*(pContribFromLibrary[19]*pSizeOfMovement))
(int) (Math.cos(1*Math.PI/180)*(pContribFromLibrary[20]*pSizeOfMovement))
(int) (Math.cos(12*Math.PI/180)*(pContribFromLibrary[21]*pSizeOfMovement))
(int) (Math.cos(25*Math.PI/180)*(pContribFromLibrary[22]*pSizeOfMovement))

```

```

-
(int) (Math.cos (38*Math.PI/180) * (pContribFromLibrary [23] *pSizeOfMovement))
-
(int) (Math.cos (51*Math.PI/180) * (pContribFromLibrary [24] *pSizeOfMovement))
-
(int) (Math.cos (64*Math.PI/180) * (pContribFromLibrary [25] *pSizeOfMovement))
-
(int) (Math.cos (77*Math.PI/180) * (pContribFromLibrary [26] *pSizeOfMovement));

    yPContig=yCircle
        - (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (76*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (62*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
-
(int) (Math.sin (48*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
-
(int) (Math.sin (34*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
-
(int) (Math.sin (20*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
-
(int) (Math.sin (6*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.sin (8*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.sin (22*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
+ (int) (Math.sin (36*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
+ (int) (Math.sin (49*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
+ (int) (Math.sin (62*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
+ (int) (Math.sin (75*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
+ (int) (Math.sin (88*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
+ (int) (Math.sin (79*Math.PI/180) * (pContribFromLibrary [14] *pSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))
+ (int) (Math.sin (53*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))
+ (int) (Math.sin (40*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement))
+ (int) (Math.sin (27*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement))
+ (int) (Math.sin (14*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement))
+ (int) (Math.sin (1*Math.PI/180) * (pContribFromLibrary [20] *pSizeOfMovement))
-
(int) (Math.sin (12*Math.PI/180) * (pContribFromLibrary [21] *pSizeOfMovement))
-
(int) (Math.sin (25*Math.PI/180) * (pContribFromLibrary [22] *pSizeOfMovement))
-
(int) (Math.sin (38*Math.PI/180) * (pContribFromLibrary [23] *pSizeOfMovement))

```

```

-
(int) (Math.sin(51*Math.PI/180) * (pContribFromLibrary[24] * pSizeOfMovement))
-
(int) (Math.sin(64*Math.PI/180) * (pContribFromLibrary[25] * pSizeOfMovement))
-
(int) (Math.sin(77*Math.PI/180) * (pContribFromLibrary[26] * pSizeOfMovement));

    xEContig=xCircle

+ (int) (Math.cos(76*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(62*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(48*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(34*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.cos(20*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
    + (int) (Math.cos(6*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.cos(8*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.cos(22*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
    + (int) (Math.cos(36*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
    + (int) (Math.cos(49*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
+ (int) (Math.cos(62*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.cos(75*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
+ (int) (Math.cos(88*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
-
(int) (Math.cos(79*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
-
(int) (Math.cos(53*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
-
(int) (Math.cos(40*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
-
(int) (Math.cos(27*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))
-
(int) (Math.cos(14*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
-
(int) (Math.cos(1*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
-
(int) (Math.cos(12*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement))
-
(int) (Math.cos(25*Math.PI/180) * (eContribFromLibrary[22] * eSizeOfMovement))
-
(int) (Math.cos(38*Math.PI/180) * (eContribFromLibrary[23] * eSizeOfMovement))
-
(int) (Math.cos(51*Math.PI/180) * (eContribFromLibrary[24] * eSizeOfMovement))

```

```

-
(int) (Math.cos (64*Math.PI/180) * (eContribFromLibrary [25] *eSizeOfMovement))
-
(int) (Math.cos (77*Math.PI/180) * (eContribFromLibrary [26] *eSizeOfMovement));

yEContig=yCircle
- (int) (eContribFromLibrary [0] *eSizeOfMovement)
-
(int) (Math.sin (76*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
-
(int) (Math.sin (62*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
-
(int) (Math.sin (48*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
-
(int) (Math.sin (34*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
-
(int) (Math.sin (20*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))
-
(int) (Math.sin (6*Math.PI/180) * (eContribFromLibrary [6] *eSizeOfMovement))
+ (int) (Math.sin (8*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
+ (int) (Math.sin (22*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
+ (int) (Math.sin (36*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))
+ (int) (Math.sin (49*Math.PI/180) * (eContribFromLibrary [10] *eSizeOfMovement))
+ (int) (Math.sin (62*Math.PI/180) * (eContribFromLibrary [11] *eSizeOfMovement))
+ (int) (Math.sin (75*Math.PI/180) * (eContribFromLibrary [12] *eSizeOfMovement))
+ (int) (Math.sin (88*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement))
+ (int) (Math.sin (79*Math.PI/180) * (eContribFromLibrary [14] *eSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (eContribFromLibrary [15] *eSizeOfMovement))
+ (int) (Math.sin (53*Math.PI/180) * (eContribFromLibrary [16] *eSizeOfMovement))
+ (int) (Math.sin (40*Math.PI/180) * (eContribFromLibrary [17] *eSizeOfMovement))
+ (int) (Math.sin (27*Math.PI/180) * (eContribFromLibrary [18] *eSizeOfMovement))
+ (int) (Math.sin (14*Math.PI/180) * (eContribFromLibrary [19] *eSizeOfMovement))
+ (int) (Math.sin (1*Math.PI/180) * (eContribFromLibrary [20] *eSizeOfMovement))
-
(int) (Math.sin (12*Math.PI/180) * (eContribFromLibrary [21] *eSizeOfMovement))
-
(int) (Math.sin (25*Math.PI/180) * (eContribFromLibrary [22] *eSizeOfMovement))
-
(int) (Math.sin (38*Math.PI/180) * (eContribFromLibrary [23] *eSizeOfMovement))
-
(int) (Math.sin (51*Math.PI/180) * (eContribFromLibrary [24] *eSizeOfMovement))
-
(int) (Math.sin (64*Math.PI/180) * (eContribFromLibrary [25] *eSizeOfMovement))

```

```

(int) (Math.sin(77*Math.PI/180) * (eContribFromLibrary[26] * eSizeOfMovement));
    xWContig=xCircle

+ (int) (Math.cos(76*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(62*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(48*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(34*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(20*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))

    + (int) (Math.cos(6*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.cos(8*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.cos(22*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))

    + (int) (Math.cos(36*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
    + (int) (Math.cos(49*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))

+ (int) (Math.cos(62*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
+ (int) (Math.cos(75*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
+ (int) (Math.cos(88*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
(int) (Math.cos(79*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
(int) (Math.cos(66*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))
(int) (Math.cos(53*Math.PI/180) * (wContribFromLibrary[16] * wSizeOfMovement))
(int) (Math.cos(40*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
(int) (Math.cos(27*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
(int) (Math.cos(14*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
(int) (Math.cos(1*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))
(int) (Math.cos(12*Math.PI/180) * (wContribFromLibrary[21] * wSizeOfMovement))
(int) (Math.cos(25*Math.PI/180) * (wContribFromLibrary[22] * wSizeOfMovement))
(int) (Math.cos(38*Math.PI/180) * (wContribFromLibrary[23] * wSizeOfMovement))
(int) (Math.cos(51*Math.PI/180) * (wContribFromLibrary[24] * wSizeOfMovement))
(int) (Math.cos(64*Math.PI/180) * (wContribFromLibrary[25] * wSizeOfMovement))
(int) (Math.cos(77*Math.PI/180) * (wContribFromLibrary[26] * wSizeOfMovement));

    yWContig=yCircle

```

```

        - (int) (wContribFromLibrary[0] * wSizeOfMovement)
    -
    (int) (Math.sin(76*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
    -
    (int) (Math.sin(62*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
    -
    (int) (Math.sin(48*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
    -
    (int) (Math.sin(34*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
    -
    (int) (Math.sin(20*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
    -
    (int) (Math.sin(6*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
    + (int) (Math.sin(8*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
    + (int) (Math.sin(22*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
    + (int) (Math.sin(36*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
    + (int) (Math.sin(49*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
    + (int) (Math.sin(62*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
    + (int) (Math.sin(75*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
    + (int) (Math.sin(88*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
    + (int) (Math.sin(79*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
    + (int) (Math.sin(66*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))
    + (int) (Math.sin(53*Math.PI/180) * (wContribFromLibrary[16] * wSizeOfMovement))
    + (int) (Math.sin(40*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
    + (int) (Math.sin(27*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
    + (int) (Math.sin(14*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
    + (int) (Math.sin(1*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))
    -
    (int) (Math.sin(12*Math.PI/180) * (wContribFromLibrary[21] * wSizeOfMovement))
    -
    (int) (Math.sin(25*Math.PI/180) * (wContribFromLibrary[22] * wSizeOfMovement))
    -
    (int) (Math.sin(38*Math.PI/180) * (wContribFromLibrary[23] * wSizeOfMovement))
    -
    (int) (Math.sin(51*Math.PI/180) * (wContribFromLibrary[24] * wSizeOfMovement))
    -
    (int) (Math.sin(64*Math.PI/180) * (wContribFromLibrary[25] * wSizeOfMovement))
    -
    (int) (Math.sin(77*Math.PI/180) * (wContribFromLibrary[26] * wSizeOfMovement));
    } // end 27

```

```

// if numberOfLibraries is 28, twenty-four 13-degree pies & four 12-degree
pies

```

```

else if ( 28 == numberOfLibraries)
{
    xPContig=xCircle

+ (int) (Math.cos (77*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
+ (int) (Math.cos (64*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
+ (int) (Math.cos (51*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
+ (int) (Math.cos (38*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
+ (int) (Math.cos (25*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))

        + (int) (Math.cos (12*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.cos (1*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.cos (14*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))

        + (int) (Math.cos (27*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
        + (int) (Math.cos (40*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))

+ (int) (Math.cos (53*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
+ (int) (Math.cos (66*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
+ (int) (Math.cos (79*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
-
(int) (Math.cos (88*Math.PI/180) * (pContribFromLibrary [14] *pSizeOfMovement))
-
(int) (Math.cos (75*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))
-
(int) (Math.cos (62*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))
-
(int) (Math.cos (49*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement))
-
(int) (Math.cos (23*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (pContribFromLibrary [20] *pSizeOfMovement))
-
(int) (Math.cos (3*Math.PI/180) * (pContribFromLibrary [21] *pSizeOfMovement))
-
(int) (Math.cos (16*Math.PI/180) * (pContribFromLibrary [22] *pSizeOfMovement))
-
(int) (Math.cos (29*Math.PI/180) * (pContribFromLibrary [23] *pSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (pContribFromLibrary [24] *pSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary [25] *pSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (pContribFromLibrary [26] *pSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (pContribFromLibrary [27] *pSizeOfMovement));

```

```

yPContig=yCircle
      -(int) (pContribFromLibrary[0] *pSizeOfMovement)
-
(int) (Math.sin(77*Math.PI/180) * (pContribFromLibrary[1] *pSizeOfMovement))
-
(int) (Math.sin(64*Math.PI/180) * (pContribFromLibrary[2] *pSizeOfMovement))
-
(int) (Math.sin(51*Math.PI/180) * (pContribFromLibrary[3] *pSizeOfMovement))
-
(int) (Math.sin(38*Math.PI/180) * (pContribFromLibrary[4] *pSizeOfMovement))
-
(int) (Math.sin(25*Math.PI/180) * (pContribFromLibrary[5] *pSizeOfMovement))
-
(int) (Math.sin(12*Math.PI/180) * (pContribFromLibrary[6] *pSizeOfMovement))
+
(int) (Math.sin(1*Math.PI/180) * (pContribFromLibrary[7] *pSizeOfMovement))
+
(int) (Math.sin(14*Math.PI/180) * (pContribFromLibrary[8] *pSizeOfMovement))
+
(int) (Math.sin(27*Math.PI/180) * (pContribFromLibrary[9] *pSizeOfMovement))
+
(int) (Math.sin(40*Math.PI/180) * (pContribFromLibrary[10] *pSizeOfMovement))
+
(int) (Math.sin(53*Math.PI/180) * (pContribFromLibrary[11] *pSizeOfMovement))
+
(int) (Math.sin(66*Math.PI/180) * (pContribFromLibrary[12] *pSizeOfMovement))
+
(int) (Math.sin(79*Math.PI/180) * (pContribFromLibrary[13] *pSizeOfMovement))
+
(int) (Math.sin(88*Math.PI/180) * (pContribFromLibrary[14] *pSizeOfMovement))
+
(int) (Math.sin(75*Math.PI/180) * (pContribFromLibrary[15] *pSizeOfMovement))
+
(int) (Math.sin(62*Math.PI/180) * (pContribFromLibrary[16] *pSizeOfMovement))
+
(int) (Math.sin(49*Math.PI/180) * (pContribFromLibrary[17] *pSizeOfMovement))
+
(int) (Math.sin(36*Math.PI/180) * (pContribFromLibrary[18] *pSizeOfMovement))
+
(int) (Math.sin(23*Math.PI/180) * (pContribFromLibrary[19] *pSizeOfMovement))
+
(int) (Math.sin(10*Math.PI/180) * (pContribFromLibrary[20] *pSizeOfMovement))
-
(int) (Math.sin(3*Math.PI/180) * (pContribFromLibrary[21] *pSizeOfMovement))
-
(int) (Math.sin(16*Math.PI/180) * (pContribFromLibrary[22] *pSizeOfMovement))
-
(int) (Math.sin(29*Math.PI/180) * (pContribFromLibrary[23] *pSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (pContribFromLibrary[24] *pSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (pContribFromLibrary[25] *pSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (pContribFromLibrary[26] *pSizeOfMovement))
-
(int) (Math.sin(78*Math.PI/180) * (pContribFromLibrary[27] *pSizeOfMovement));

```

```

    xEContig=xCircle

+ (int) (Math.cos (77*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
+ (int) (Math.cos (64*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
+ (int) (Math.cos (51*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
+ (int) (Math.cos (38*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
+ (int) (Math.cos (25*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))

    + (int) (Math.cos (12*Math.PI/180) * (eContribFromLibrary [6] *eSizeOfMovement))
+ (int) (Math.cos (1*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
+ (int) (Math.cos (14*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))

    + (int) (Math.cos (27*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))
    + (int) (Math.cos (40*Math.PI/180) * (eContribFromLibrary [10] *eSizeOfMovement))
+ (int) (Math.cos (53*Math.PI/180) * (eContribFromLibrary [11] *eSizeOfMovement))
+ (int) (Math.cos (66*Math.PI/180) * (eContribFromLibrary [12] *eSizeOfMovement))
+ (int) (Math.cos (79*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement))
-
(int) (Math.cos (88*Math.PI/180) * (eContribFromLibrary [14] *eSizeOfMovement))
-
(int) (Math.cos (75*Math.PI/180) * (eContribFromLibrary [15] *eSizeOfMovement))
-
(int) (Math.cos (62*Math.PI/180) * (eContribFromLibrary [16] *eSizeOfMovement))
-
(int) (Math.cos (49*Math.PI/180) * (eContribFromLibrary [17] *eSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (eContribFromLibrary [18] *eSizeOfMovement))
-
(int) (Math.cos (23*Math.PI/180) * (eContribFromLibrary [19] *eSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (eContribFromLibrary [20] *eSizeOfMovement))
-
(int) (Math.cos (3*Math.PI/180) * (eContribFromLibrary [21] *eSizeOfMovement))
-
(int) (Math.cos (16*Math.PI/180) * (eContribFromLibrary [22] *eSizeOfMovement))
-
(int) (Math.cos (29*Math.PI/180) * (eContribFromLibrary [23] *eSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (eContribFromLibrary [24] *eSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (eContribFromLibrary [25] *eSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (eContribFromLibrary [26] *eSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (eContribFromLibrary [27] *eSizeOfMovement));

    yEContig=yCircle

```

```

        - (int) (eContribFromLibrary[0] * eSizeOfMovement)
    -
    (int) (Math.sin(77*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
    -
    (int) (Math.sin(64*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
    -
    (int) (Math.sin(51*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
    -
    (int) (Math.sin(38*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
    -
    (int) (Math.sin(25*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
    -
    (int) (Math.sin(12*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
    + (int) (Math.sin(1*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
    + (int) (Math.sin(14*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
    + (int) (Math.sin(27*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
    + (int) (Math.sin(40*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
    + (int) (Math.sin(53*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
    + (int) (Math.sin(66*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
    + (int) (Math.sin(79*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
    + (int) (Math.sin(88*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
    + (int) (Math.sin(75*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
    + (int) (Math.sin(62*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
    + (int) (Math.sin(49*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
    + (int) (Math.sin(36*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))
    + (int) (Math.sin(23*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
    + (int) (Math.sin(10*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
    -
    (int) (Math.sin(3*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement))
    -
    (int) (Math.sin(16*Math.PI/180) * (eContribFromLibrary[22] * eSizeOfMovement))
    -
    (int) (Math.sin(29*Math.PI/180) * (eContribFromLibrary[23] * eSizeOfMovement))
    -
    (int) (Math.sin(42*Math.PI/180) * (eContribFromLibrary[24] * eSizeOfMovement))
    -
    (int) (Math.sin(54*Math.PI/180) * (eContribFromLibrary[25] * eSizeOfMovement))
    -
    (int) (Math.sin(66*Math.PI/180) * (eContribFromLibrary[26] * eSizeOfMovement))
    -
    (int) (Math.sin(78*Math.PI/180) * (eContribFromLibrary[27] * eSizeOfMovement));

    xWContig=xCircle

```

```

+ (int) (Math.cos (77*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
+ (int) (Math.cos (64*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
+ (int) (Math.cos (51*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
+ (int) (Math.cos (38*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
+ (int) (Math.cos (25*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
      + (int) (Math.cos (12*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
+ (int) (Math.cos (1*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
+ (int) (Math.cos (14*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
      + (int) (Math.cos (27*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
      + (int) (Math.cos (40*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
+ (int) (Math.cos (53*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
+ (int) (Math.cos (66*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
+ (int) (Math.cos (79*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
-
(int) (Math.cos (88*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
-
(int) (Math.cos (75*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
-
(int) (Math.cos (62*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))
-
(int) (Math.cos (49*Math.PI/180) * (wContribFromLibrary [17] *wSizeOfMovement))
-
(int) (Math.cos (36*Math.PI/180) * (wContribFromLibrary [18] *wSizeOfMovement))
-
(int) (Math.cos (23*Math.PI/180) * (wContribFromLibrary [19] *wSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (wContribFromLibrary [20] *wSizeOfMovement))
-
(int) (Math.cos (3*Math.PI/180) * (wContribFromLibrary [21] *wSizeOfMovement))
-
(int) (Math.cos (16*Math.PI/180) * (wContribFromLibrary [22] *wSizeOfMovement))
-
(int) (Math.cos (29*Math.PI/180) * (wContribFromLibrary [23] *wSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (wContribFromLibrary [24] *wSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary [25] *wSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (wContribFromLibrary [26] *wSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (wContribFromLibrary [27] *wSizeOfMovement));

yWContig=yCircle
      - (int) (wContribFromLibrary [0] *wSizeOfMovement)

```

```

-
(int) (Math.sin(77*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
-
(int) (Math.sin(64*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
-
(int) (Math.sin(51*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
-
(int) (Math.sin(38*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
-
(int) (Math.sin(25*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
-
(int) (Math.sin(12*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.sin(1*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.sin(14*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
+ (int) (Math.sin(27*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
+ (int) (Math.sin(40*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
+ (int) (Math.sin(53*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
+ (int) (Math.sin(66*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
+ (int) (Math.sin(79*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
+ (int) (Math.sin(88*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
+ (int) (Math.sin(75*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))
+ (int) (Math.sin(62*Math.PI/180) * (wContribFromLibrary[16] * wSizeOfMovement))
+ (int) (Math.sin(49*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
+ (int) (Math.sin(36*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
+ (int) (Math.sin(23*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
+ (int) (Math.sin(10*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))
-
(int) (Math.sin(3*Math.PI/180) * (wContribFromLibrary[21] * wSizeOfMovement))
-
(int) (Math.sin(16*Math.PI/180) * (wContribFromLibrary[22] * wSizeOfMovement))
-
(int) (Math.sin(29*Math.PI/180) * (wContribFromLibrary[23] * wSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (wContribFromLibrary[24] * wSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (wContribFromLibrary[25] * wSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (wContribFromLibrary[26] * wSizeOfMovement))
-
(int) (Math.sin(78*Math.PI/180) * (wContribFromLibrary[27] * wSizeOfMovement));
    } // end 28

/**

```

```
// REMOVE COMMENT LATER
```

```
// if numberOfLibraries is 29, twelve 13-degree, seventeen 12-degree  
else if ( 29 == numberOfLibraries)
```

```
{
```

```
    xPContig=xCircle
```

```
+ (int) (Math.cos (77*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
```

```
+ (int) (Math.cos (64*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
```

```
+ (int) (Math.cos (51*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
```

```
+ (int) (Math.cos (38*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
```

```
+ (int) (Math.cos (25*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
```

```
    + (int) (Math.cos (12*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
```

```
+ (int) (Math.cos (1*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
```

```
+ (int) (Math.cos (14*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
```

```
    + (int) (Math.cos (27*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
```

```
    + (int) (Math.cos (40*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
```

```
+ (int) (Math.cos (53*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
```

```
+ (int) (Math.cos (66*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
```

```
+ (int) (Math.cos (78*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
```

```
-  
(int) (Math.cos (78*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))
```

```
-  
(int) (Math.cos (66*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))
```

```
-  
(int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement))
```

```
-  
(int) (Math.cos (42*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement))
```

```
-  
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement))
```

```
-  
(int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary [20] *pSizeOfMovement))
```

```
-  
(int) (Math.cos (6*Math.PI/180) * (pContribFromLibrary [21] *pSizeOfMovement))
```

```
-  
(int) (Math.cos (6*Math.PI/180) * (pContribFromLibrary [22] *pSizeOfMovement))
```

```
-  
(int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary [23] *pSizeOfMovement))
```

```
-  
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [24] *pSizeOfMovement))
```

```
-  
(int) (Math.cos (42*Math.PI/180) * (pContribFromLibrary [25] *pSizeOfMovement))
```

```
-  
(int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary [26] *pSizeOfMovement))
```

```

(int) (Math.cos (66*Math.PI/180) * (pContribFromLibrary [27] *pSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (pContribFromLibrary [28] *pSizeOfMovement));

    yPContig=yCircle
        - (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (77*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (64*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
-
(int) (Math.sin (51*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
-
(int) (Math.sin (38*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
-
(int) (Math.sin (25*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
-
(int) (Math.sin (12*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.sin (1*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.sin (14*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
+ (int) (Math.sin (27*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
+ (int) (Math.sin (40*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
+ (int) (Math.sin (53*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
    + (int) (pContribFromLibrary [14] *pSizeOfMovement)
+ (int) (Math.sin (78*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (pContribFromLibrary [20] *pSizeOfMovement))
+ (int) (Math.sin (6*Math.PI/180) * (pContribFromLibrary [21] *pSizeOfMovement))
-
(int) (Math.sin (6*Math.PI/180) * (pContribFromLibrary [22] *pSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (pContribFromLibrary [23] *pSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [24] *pSizeOfMovement))
-
(int) (Math.sin (42*Math.PI/180) * (pContribFromLibrary [25] *pSizeOfMovement))

```

```

-
(int) (Math.sin(54*Math.PI/180) * (pContribFromLibrary[26] * pSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (pContribFromLibrary[27] * pSizeOfMovement))
-
(int) (Math.sin(78*Math.PI/180) * (pContribFromLibrary[28] * pSizeOfMovement));

    xEContig=xCircle

+ (int) (Math.cos(77*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(64*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(51*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(38*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.cos(25*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
    + (int) (Math.cos(12*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.cos(1*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.cos(14*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
    + (int) (Math.cos(27*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
    + (int) (Math.cos(40*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
+ (int) (Math.cos(53*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.cos(66*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))

-
(int) (Math.cos(78*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
-
(int) (Math.cos(54*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
-
(int) (Math.cos(42*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
-
(int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
-
(int) (Math.cos(6*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement))
-
(int) (Math.cos(6*Math.PI/180) * (eContribFromLibrary[22] * eSizeOfMovement))
-
(int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[23] * eSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[24] * eSizeOfMovement))
-
(int) (Math.cos(42*Math.PI/180) * (eContribFromLibrary[25] * eSizeOfMovement))

```

```

-
(int) (Math.cos (54*Math.PI/180) * (eContribFromLibrary [26] *eSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (eContribFromLibrary [27] *eSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (eContribFromLibrary [28] *eSizeOfMovement));

    yEContig=yCircle
        - (int) (eContribFromLibrary [0] *eSizeOfMovement)
-
(int) (Math.sin (77*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
-
(int) (Math.sin (64*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
-
(int) (Math.sin (51*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
-
(int) (Math.sin (38*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
-
(int) (Math.sin (25*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))
-
(int) (Math.sin (12*Math.PI/180) * (eContribFromLibrary [6] *eSizeOfMovement))
+ (int) (Math.sin (1*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
+ (int) (Math.sin (14*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
+ (int) (Math.sin (27*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))
+ (int) (Math.sin (40*Math.PI/180) * (eContribFromLibrary [10] *eSizeOfMovement))
+ (int) (Math.sin (53*Math.PI/180) * (eContribFromLibrary [11] *eSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (eContribFromLibrary [12] *eSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement))
    + (int) (eContribFromLibrary [14] *eSizeOfMovement)
+ (int) (Math.sin (78*Math.PI/180) * (eContribFromLibrary [15] *eSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (eContribFromLibrary [16] *eSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (eContribFromLibrary [17] *eSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (eContribFromLibrary [18] *eSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (eContribFromLibrary [19] *eSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (eContribFromLibrary [20] *eSizeOfMovement))
+ (int) (Math.sin (6*Math.PI/180) * (eContribFromLibrary [21] *eSizeOfMovement))
-
(int) (Math.sin (6*Math.PI/180) * (eContribFromLibrary [22] *eSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (eContribFromLibrary [23] *eSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (eContribFromLibrary [24] *eSizeOfMovement))

```

```

-
(int) (Math.sin(42*Math.PI/180) * (eContribFromLibrary[25] * eSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (eContribFromLibrary[26] * eSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (eContribFromLibrary[27] * eSizeOfMovement))
-
(int) (Math.sin(78*Math.PI/180) * (eContribFromLibrary[28] * eSizeOfMovement));

    xWContig=xCircle

+ (int) (Math.cos(77*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(64*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(51*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(38*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(25*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
    + (int) (Math.cos(12*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.cos(1*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.cos(14*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
    + (int) (Math.cos(27*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
    + (int) (Math.cos(40*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
+ (int) (Math.cos(53*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
+ (int) (Math.cos(66*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))

-
(int) (Math.cos(78*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (wContribFromLibrary[16] * wSizeOfMovement))
-
(int) (Math.cos(54*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
-
(int) (Math.cos(42*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
-
(int) (Math.cos(18*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))
-
(int) (Math.cos(6*Math.PI/180) * (wContribFromLibrary[21] * wSizeOfMovement))
-
(int) (Math.cos(6*Math.PI/180) * (wContribFromLibrary[22] * wSizeOfMovement))
-
(int) (Math.cos(18*Math.PI/180) * (wContribFromLibrary[23] * wSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[24] * wSizeOfMovement))

```

```

-
(int) (Math.cos (42*Math.PI/180) * (wContribFromLibrary [25] *wSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary [26] *wSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (wContribFromLibrary [27] *wSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (wContribFromLibrary [28] *wSizeOfMovement));

    yWContig=yCircle
        - (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (77*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
-
(int) (Math.sin (64*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
-
(int) (Math.sin (51*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
-
(int) (Math.sin (38*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
-
(int) (Math.sin (25*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
-
(int) (Math.sin (12*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
+ (int) (Math.sin (1*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
+ (int) (Math.sin (14*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
+ (int) (Math.sin (27*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
+ (int) (Math.sin (40*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
+ (int) (Math.sin (53*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
    + (int) (wContribFromLibrary [14] *wSizeOfMovement)
+ (int) (Math.sin (78*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary [17] *wSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (wContribFromLibrary [18] *wSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary [19] *wSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary [20] *wSizeOfMovement))
+ (int) (Math.sin (6*Math.PI/180) * (wContribFromLibrary [21] *wSizeOfMovement))
-
(int) (Math.sin (6*Math.PI/180) * (wContribFromLibrary [22] *wSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary [23] *wSizeOfMovement))

```

```

-
(int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[24] * wSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (wContribFromLibrary[25] * wSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (wContribFromLibrary[26] * wSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (wContribFromLibrary[27] * wSizeOfMovement))
-
(int) (Math.sin(78*Math.PI/180) * (wContribFromLibrary[28] * wSizeOfMovement));
    } // end 29

    // if numberOfLibraries is 30, all 12-degree pies
    else if ( 30 == numberOfLibraries)
    {
        xPContig=xCircle

+ (int) (Math.cos(78*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(66*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))

        + (int) (Math.cos(18*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))

        + (int) (Math.cos(18*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
        + (int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement))
+ (int) (Math.cos(66*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))

-
(int) (Math.cos(78*Math.PI/180) * (pContribFromLibrary[16] * pSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (pContribFromLibrary[17] * pSizeOfMovement))
-
(int) (Math.cos(54*Math.PI/180) * (pContribFromLibrary[18] * pSizeOfMovement))
-
(int) (Math.cos(42*Math.PI/180) * (pContribFromLibrary[19] * pSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (pContribFromLibrary[20] * pSizeOfMovement))
-
(int) (Math.cos(18*Math.PI/180) * (pContribFromLibrary[21] * pSizeOfMovement))

```

```

-
(int) (Math.cos (6*Math.PI/180) * (pContribFromLibrary [22] *pSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (pContribFromLibrary [23] *pSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary [24] *pSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [25] *pSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (pContribFromLibrary [26] *pSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary [27] *pSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (pContribFromLibrary [28] *pSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (pContribFromLibrary [29] *pSizeOfMovement));

yPContig=yCircle
- (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (78*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (66*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
-
(int) (Math.sin (54*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
-
(int) (Math.sin (42*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
-
(int) (Math.sin (6*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.sin (6*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (pContribFromLibrary [14] *pSizeOfMovement))
+ (int) (pContribFromLibrary [15] *pSizeOfMovement)
+ (int) (Math.sin (78*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement))

```

```

+ (int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[20] * pSizeOfMovement))
+ (int) (Math.sin(18*Math.PI/180) * (pContribFromLibrary[21] * pSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (pContribFromLibrary[22] * pSizeOfMovement))
-
(int) (Math.sin(6*Math.PI/180) * (pContribFromLibrary[23] * pSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (pContribFromLibrary[24] * pSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[25] * pSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (pContribFromLibrary[26] * pSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (pContribFromLibrary[27] * pSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (pContribFromLibrary[28] * pSizeOfMovement))
-
(int) (Math.sin(78*Math.PI/180) * (pContribFromLibrary[29] * pSizeOfMovement));

    xEContig=xCircle

+ (int) (Math.cos(78*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(66*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))

    + (int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))

    + (int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
    + (int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))

+ (int) (Math.cos(42*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
+ (int) (Math.cos(66*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))

-
(int) (Math.cos(78*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
-
(int) (Math.cos(54*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))

```

```

-
(int) (Math.cos (42*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (eContribFromLibrary[22] * eSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (eContribFromLibrary[23] * eSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (eContribFromLibrary[24] * eSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary[25] * eSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (eContribFromLibrary[26] * eSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (eContribFromLibrary[27] * eSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (eContribFromLibrary[28] * eSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (eContribFromLibrary[29] * eSizeOfMovement));

yEContig=yCircle
- (int) (eContribFromLibrary[0] * eSizeOfMovement)
-
(int) (Math.sin (78*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
-
(int) (Math.sin (66*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
-
(int) (Math.sin (54*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
-
(int) (Math.sin (42*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
-
(int) (Math.sin (6*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.sin (6*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
+ (int) (eContribFromLibrary[15] * eSizeOfMovement)
+ (int) (Math.sin (78*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))

```

```

+ (int) (Math.sin(66*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))
+ (int) (Math.sin(42*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
+ (int) (Math.sin(18*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (eContribFromLibrary[22] * eSizeOfMovement))
-
(int) (Math.sin(6*Math.PI/180) * (eContribFromLibrary[23] * eSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (eContribFromLibrary[24] * eSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[25] * eSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (eContribFromLibrary[26] * eSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (eContribFromLibrary[27] * eSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (eContribFromLibrary[28] * eSizeOfMovement))
-
(int) (Math.sin(78*Math.PI/180) * (eContribFromLibrary[29] * eSizeOfMovement));

```

xWContig=xCircle

```

+ (int) (Math.cos(78*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(66*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
+ (int) (Math.cos(18*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.cos(6*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
+ (int) (Math.cos(18*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
+ (int) (Math.cos(30*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
+ (int) (Math.cos(66*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))

```

```

-
(int) (Math.cos (78*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (wContribFromLibrary [17] *wSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary [18] *wSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (wContribFromLibrary [19] *wSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (wContribFromLibrary [20] *wSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary [21] *wSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (wContribFromLibrary [22] *wSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (wContribFromLibrary [23] *wSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary [24] *wSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (wContribFromLibrary [25] *wSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (wContribFromLibrary [26] *wSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary [27] *wSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (wContribFromLibrary [28] *wSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (wContribFromLibrary [29] *wSizeOfMovement));

yWContig=yCircle
- (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (78*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
-
(int) (Math.sin (66*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
-
(int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
-
(int) (Math.sin (42*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
-
(int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
-
(int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
-
(int) (Math.sin (6*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
+ (int) (Math.sin (6*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))

```

```

+ (int) (Math.sin(66*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
    + (int) (wContribFromLibrary[15] * wSizeOfMovement)
+ (int) (Math.sin(78*Math.PI/180) * (wContribFromLibrary[16] * wSizeOfMovement))
+ (int) (Math.sin(66*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
+ (int) (Math.sin(42*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))
+ (int) (Math.sin(18*Math.PI/180) * (wContribFromLibrary[21] * wSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (wContribFromLibrary[22] * wSizeOfMovement))
-
(int) (Math.sin(6*Math.PI/180) * (wContribFromLibrary[23] * wSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (wContribFromLibrary[24] * wSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[25] * wSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (wContribFromLibrary[26] * wSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (wContribFromLibrary[27] * wSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (wContribFromLibrary[28] * wSizeOfMovement))
-
(int) (Math.sin(78*Math.PI/180) * (wContribFromLibrary[29] * wSizeOfMovement));
    } // end 30
// remove comment later
*/
    } // end calcXY_27to30()
//*****
// remove comment later
/**
//*****

void calcXY_31to33()
//*****

{
    // if numberOfLibraries is 31, 12-degree & 11-degree slices
    if ( 31 == numberOfLibraries)
    {
        xPContig=xCircle

+ (int) (Math.cos(79*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(68*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.cos(57*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))

```

```

+ (int) (Math.cos (46*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
+ (int) (Math.cos (35*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
    + (int) (Math.cos (24*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
+ (int) (Math.cos (13*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
+ (int) (Math.cos (2*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
    + (int) (Math.cos (9*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
    + (int) (Math.cos (20*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
+ (int) (Math.cos (31*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
+ (int) (Math.cos (42*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
+ (int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
+ (int) (Math.cos (66*Math.PI/180) * (pContribFromLibrary [14] *pSizeOfMovement))
+ (int) (Math.cos (78*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))

-
(int) (Math.cos (78*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (pContribFromLibrary [20] *pSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [21] *pSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary [22] *pSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (pContribFromLibrary [23] *pSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (pContribFromLibrary [24] *pSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary [25] *pSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [26] *pSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (pContribFromLibrary [27] *pSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary [28] *pSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (pContribFromLibrary [29] *pSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (pContribFromLibrary [30] *pSizeOfMovement));

yPContig=yCircle
    - (int) (pContribFromLibrary [0] *pSizeOfMovement)

```

```

-
(int) (Math.sin(79*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
-
(int) (Math.sin(68*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
-
(int) (Math.sin(57*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
-
(int) (Math.sin(46*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
-
(int) (Math.sin(35*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
-
(int) (Math.sin(24*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
-
(int) (Math.sin(13*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
-
(int) (Math.sin(2*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))
+ (int) (Math.sin(9*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
+ (int) (Math.sin(20*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))
+ (int) (Math.sin(31*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
+ (int) (Math.sin(42*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
+ (int) (Math.sin(66*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))
+ (int) (Math.sin(78*Math.PI/180) * (pContribFromLibrary[15] * pSizeOfMovement))
+ (int) (pContribFromLibrary[16] * pSizeOfMovement)
+ (int) (Math.sin(78*Math.PI/180) * (pContribFromLibrary[17] * pSizeOfMovement))
+ (int) (Math.sin(66*Math.PI/180) * (pContribFromLibrary[18] * pSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (pContribFromLibrary[19] * pSizeOfMovement))
+ (int) (Math.sin(42*Math.PI/180) * (pContribFromLibrary[20] * pSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[21] * pSizeOfMovement))
+ (int) (Math.sin(18*Math.PI/180) * (pContribFromLibrary[22] * pSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (pContribFromLibrary[23] * pSizeOfMovement))
-
(int) (Math.sin(6*Math.PI/180) * (pContribFromLibrary[24] * pSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (pContribFromLibrary[25] * pSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[26] * pSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (pContribFromLibrary[27] * pSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (pContribFromLibrary[28] * pSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (pContribFromLibrary[29] * pSizeOfMovement))

```

```

(int) (Math.sin(78*Math.PI/180) * (pContribFromLibrary[30] * pSizeOfMovement));

    xEContig=xCircle

+ (int) (Math.cos(79*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(68*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(57*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(46*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.cos(35*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))

    + (int) (Math.cos(24*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.cos(13*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.cos(2*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))

    + (int) (Math.cos(9*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
    + (int) (Math.cos(20*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))

    + (int) (Math.cos(31*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
+ (int) (Math.cos(66*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))

    -
(int) (Math.cos(78*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
    -
(int) (Math.cos(66*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))
    -
(int) (Math.cos(54*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
    -
(int) (Math.cos(42*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
    -
(int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement))
    -
(int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[22] * eSizeOfMovement))
    -
(int) (Math.cos(6*Math.PI/180) * (eContribFromLibrary[23] * eSizeOfMovement))
    -
(int) (Math.cos(6*Math.PI/180) * (eContribFromLibrary[24] * eSizeOfMovement))
    -
(int) (Math.cos(18*Math.PI/180) * (eContribFromLibrary[25] * eSizeOfMovement))
    -
(int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[26] * eSizeOfMovement))

```

```

(int) (Math.cos (42*Math.PI/180) * (eContribFromLibrary[27] * eSizeOfMovement))
(int) (Math.cos (54*Math.PI/180) * (eContribFromLibrary[28] * eSizeOfMovement))
(int) (Math.cos (66*Math.PI/180) * (eContribFromLibrary[29] * eSizeOfMovement))
(int) (Math.cos (78*Math.PI/180) * (eContribFromLibrary[30] * eSizeOfMovement));

    yEContig=yCircle
        - (int) (eContribFromLibrary[0] * eSizeOfMovement)
(int) (Math.sin (79*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
(int) (Math.sin (68*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
(int) (Math.sin (57*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
(int) (Math.sin (46*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
(int) (Math.sin (35*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
(int) (Math.sin (24*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
(int) (Math.sin (13*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
(int) (Math.sin (2*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
+ (int) (Math.sin (9*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
+ (int) (Math.sin (20*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
+ (int) (Math.sin (31*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
    + (int) (eContribFromLibrary[16] * eSizeOfMovement)
+ (int) (Math.sin (78*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
+ (int) (Math.sin (30*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement))
+ (int) (Math.sin (18*Math.PI/180) * (eContribFromLibrary[22] * eSizeOfMovement))
+ (int) (Math.sin (6*Math.PI/180) * (eContribFromLibrary[23] * eSizeOfMovement))

```

```

-
(int) (Math.sin(6*Math.PI/180) * (eContribFromLibrary[24] * eSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (eContribFromLibrary[25] * eSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[26] * eSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (eContribFromLibrary[27] * eSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (eContribFromLibrary[28] * eSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (eContribFromLibrary[29] * eSizeOfMovement))
-
(int) (Math.sin(78*Math.PI/180) * (eContribFromLibrary[30] * eSizeOfMovement));

    xWContig=xCircle

+ (int) (Math.cos(79*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(68*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(57*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(46*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(35*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))

    + (int) (Math.cos(24*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.cos(13*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.cos(2*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))

    + (int) (Math.cos(9*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
    + (int) (Math.cos(20*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))

    + (int) (Math.cos(31*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
+ (int) (Math.cos(54*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
+ (int) (Math.cos(66*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
+ (int) (Math.cos(78*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))

-
(int) (Math.cos(78*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
-
(int) (Math.cos(66*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
-
(int) (Math.cos(54*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
-
(int) (Math.cos(42*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))

```

```

-
(int) (Math.cos (30*Math.PI/180) * (wContribFromLibrary [21] *wSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary [22] *wSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (wContribFromLibrary [23] *wSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (wContribFromLibrary [24] *wSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary [25] *wSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (wContribFromLibrary [26] *wSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (wContribFromLibrary [27] *wSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary [28] *wSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (wContribFromLibrary [29] *wSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (wContribFromLibrary [30] *wSizeOfMovement));

    yWContig=yCircle
        - (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (79*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
-
(int) (Math.sin (68*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
-
(int) (Math.sin (57*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
-
(int) (Math.sin (46*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
-
(int) (Math.sin (35*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
-
(int) (Math.sin (24*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
-
(int) (Math.sin (13*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
-
(int) (Math.sin (2*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
+ (int) (Math.sin (9*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
+ (int) (Math.sin (20*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
+ (int) (Math.sin (31*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
+ (int) (Math.sin (54*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
+ (int) (Math.sin (66*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
+ (int) (Math.sin (78*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
    + (int) (wContribFromLibrary [16] *wSizeOfMovement)
+ (int) (Math.sin (78*Math.PI/180) * (wContribFromLibrary [17] *wSizeOfMovement))

```

```

+ (int) (Math.sin(66*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
+ (int) (Math.sin(54*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
+ (int) (Math.sin(42*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[21] * wSizeOfMovement))
+ (int) (Math.sin(18*Math.PI/180) * (wContribFromLibrary[22] * wSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (wContribFromLibrary[23] * wSizeOfMovement))
-
(int) (Math.sin(6*Math.PI/180) * (wContribFromLibrary[24] * wSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (wContribFromLibrary[25] * wSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[26] * wSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (wContribFromLibrary[27] * wSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (wContribFromLibrary[28] * wSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (wContribFromLibrary[29] * wSizeOfMovement))
-
(int) (Math.sin(78*Math.PI/180) * (wContribFromLibrary[30] * wSizeOfMovement));
    } // end 31

    // if numberOfLibraries is 32, eight 12-degree & 24 11-degree slices
    else if ( 32 == numberOfLibraries)
    {
        xPContig=xCircle

+ (int) (Math.cos(79*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(68*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.cos(57*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (Math.cos(46*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
+ (int) (Math.cos(35*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))

        + (int) (Math.cos(24*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
+ (int) (Math.cos(13*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
+ (int) (Math.cos(2*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))

        + (int) (Math.cos(9*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
        + (int) (Math.cos(20*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))
+ (int) (Math.cos(31*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement))

```

```

+ (int) (Math.cos (53*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
+ (int) (Math.cos (64*Math.PI/180) * (pContribFromLibrary [14] *pSizeOfMovement))
+ (int) (Math.cos (75*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))
+ (int) (Math.cos (86*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))
-
(int) (Math.cos (83*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement))
-
(int) (Math.cos (72*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement))
-
(int) (Math.cos (61*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (pContribFromLibrary [20] *pSizeOfMovement))
-
(int) (Math.cos (39*Math.PI/180) * (pContribFromLibrary [21] *pSizeOfMovement))
-
(int) (Math.cos (28*Math.PI/180) * (pContribFromLibrary [22] *pSizeOfMovement))
-
(int) (Math.cos (17*Math.PI/180) * (pContribFromLibrary [23] *pSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (pContribFromLibrary [24] *pSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (pContribFromLibrary [25] *pSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (pContribFromLibrary [26] *pSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [27] *pSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (pContribFromLibrary [28] *pSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (pContribFromLibrary [29] *pSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (pContribFromLibrary [30] *pSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (pContribFromLibrary [31] *pSizeOfMovement));

    yPContig=yCircle
        - (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (79*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (68*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
-
(int) (Math.sin (57*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
-
(int) (Math.sin (46*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
-
(int) (Math.sin (35*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
-
(int) (Math.sin (24*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
-
(int) (Math.sin (13*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
-
(int) (Math.sin (2*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))

```

```

+ (int) (Math.sin(9*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
+ (int) (Math.sin(20*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))
+ (int) (Math.sin(31*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
+ (int) (Math.sin(42*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement))
+ (int) (Math.sin(53*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
+ (int) (Math.sin(64*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))
+ (int) (Math.sin(75*Math.PI/180) * (pContribFromLibrary[15] * pSizeOfMovement))
+ (int) (Math.sin(86*Math.PI/180) * (pContribFromLibrary[16] * pSizeOfMovement))
+ (int) (Math.sin(83*Math.PI/180) * (pContribFromLibrary[17] * pSizeOfMovement))
+ (int) (Math.sin(72*Math.PI/180) * (pContribFromLibrary[18] * pSizeOfMovement))
+ (int) (Math.sin(61*Math.PI/180) * (pContribFromLibrary[19] * pSizeOfMovement))
+ (int) (Math.sin(50*Math.PI/180) * (pContribFromLibrary[20] * pSizeOfMovement))
+ (int) (Math.sin(39*Math.PI/180) * (pContribFromLibrary[21] * pSizeOfMovement))
+ (int) (Math.sin(28*Math.PI/180) * (pContribFromLibrary[22] * pSizeOfMovement))
+ (int) (Math.sin(17*Math.PI/180) * (pContribFromLibrary[23] * pSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (pContribFromLibrary[24] * pSizeOfMovement))
-
(int) (Math.sin(6*Math.PI/180) * (pContribFromLibrary[25] * pSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (pContribFromLibrary[26] * pSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[27] * pSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (pContribFromLibrary[28] * pSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (pContribFromLibrary[29] * pSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (pContribFromLibrary[30] * pSizeOfMovement))
-
(int) (Math.sin(78*Math.PI/180) * (pContribFromLibrary[31] * pSizeOfMovement));

```

xEContig=xCircle

```

+ (int) (Math.cos(79*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(68*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(57*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(46*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))

```

```

+ (int) (Math.cos (35*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))
      + (int) (Math.cos (24*Math.PI/180) * (eContribFromLibrary [6] *eSizeOfMovement))
+ (int) (Math.cos (13*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
+ (int) (Math.cos (2*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
      + (int) (Math.cos (9*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))
      + (int) (Math.cos (20*Math.PI/180) * (eContribFromLibrary [10] *eSizeOfMovement))

      + (int) (Math.cos (31*Math.PI/180) * (eContribFromLibrary [11] *eSizeOfMovement))
+ (int) (Math.cos (42*Math.PI/180) * (eContribFromLibrary [12] *eSizeOfMovement))
+ (int) (Math.cos (53*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement))
+ (int) (Math.cos (64*Math.PI/180) * (eContribFromLibrary [14] *eSizeOfMovement))
+ (int) (Math.cos (75*Math.PI/180) * (eContribFromLibrary [15] *eSizeOfMovement))
+ (int) (Math.cos (86*Math.PI/180) * (eContribFromLibrary [16] *eSizeOfMovement))
-
(int) (Math.cos (83*Math.PI/180) * (eContribFromLibrary [17] *eSizeOfMovement))
-
(int) (Math.cos (72*Math.PI/180) * (eContribFromLibrary [18] *eSizeOfMovement))
-
(int) (Math.cos (61*Math.PI/180) * (eContribFromLibrary [19] *eSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (eContribFromLibrary [20] *eSizeOfMovement))
-
(int) (Math.cos (39*Math.PI/180) * (eContribFromLibrary [21] *eSizeOfMovement))
-
(int) (Math.cos (28*Math.PI/180) * (eContribFromLibrary [22] *eSizeOfMovement))
-
(int) (Math.cos (17*Math.PI/180) * (eContribFromLibrary [23] *eSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (eContribFromLibrary [24] *eSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (eContribFromLibrary [25] *eSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (eContribFromLibrary [26] *eSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [27] *eSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (eContribFromLibrary [28] *eSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (eContribFromLibrary [29] *eSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (eContribFromLibrary [30] *eSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (eContribFromLibrary [31] *eSizeOfMovement));

yEContig=yCircle

```

```

- (int) (eContribFromLibrary[0] * eSizeOfMovement)
-
(int) (Math.sin(79*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
-
(int) (Math.sin(68*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
-
(int) (Math.sin(57*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
-
(int) (Math.sin(46*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
-
(int) (Math.sin(35*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))
-
(int) (Math.sin(24*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
-
(int) (Math.sin(13*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
-
(int) (Math.sin(2*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))
+ (int) (Math.sin(9*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
+ (int) (Math.sin(20*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))
+ (int) (Math.sin(31*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.sin(42*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
+ (int) (Math.sin(53*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
+ (int) (Math.sin(64*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
+ (int) (Math.sin(75*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
+ (int) (Math.sin(86*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))
+ (int) (Math.sin(83*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
+ (int) (Math.sin(72*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))
+ (int) (Math.sin(61*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
+ (int) (Math.sin(50*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
+ (int) (Math.sin(39*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement))
+ (int) (Math.sin(28*Math.PI/180) * (eContribFromLibrary[22] * eSizeOfMovement))
+ (int) (Math.sin(17*Math.PI/180) * (eContribFromLibrary[23] * eSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (eContribFromLibrary[24] * eSizeOfMovement))
-
(int) (Math.sin(6*Math.PI/180) * (eContribFromLibrary[25] * eSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (eContribFromLibrary[26] * eSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[27] * eSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (eContribFromLibrary[28] * eSizeOfMovement))

```

```

-
(int) (Math.sin(54*Math.PI/180) * (eContribFromLibrary[29] * eSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (eContribFromLibrary[30] * eSizeOfMovement))
-
(int) (Math.sin(78*Math.PI/180) * (eContribFromLibrary[31] * eSizeOfMovement));

    xWContig=xCircle

+ (int) (Math.cos(79*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(68*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(57*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(46*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(35*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
    + (int) (Math.cos(24*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.cos(13*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.cos(2*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
    + (int) (Math.cos(9*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
    + (int) (Math.cos(20*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))

    + (int) (Math.cos(31*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
+ (int) (Math.cos(53*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
+ (int) (Math.cos(64*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
+ (int) (Math.cos(75*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))
+ (int) (Math.cos(86*Math.PI/180) * (wContribFromLibrary[16] * wSizeOfMovement))
-
(int) (Math.cos(83*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
-
(int) (Math.cos(72*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
-
(int) (Math.cos(61*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
-
(int) (Math.cos(50*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))
-
(int) (Math.cos(39*Math.PI/180) * (wContribFromLibrary[21] * wSizeOfMovement))
-
(int) (Math.cos(28*Math.PI/180) * (wContribFromLibrary[22] * wSizeOfMovement))
-
(int) (Math.cos(17*Math.PI/180) * (wContribFromLibrary[23] * wSizeOfMovement))
-
(int) (Math.cos(6*Math.PI/180) * (wContribFromLibrary[24] * wSizeOfMovement))

```

```

-
(int) (Math.cos (6*Math.PI/180) * (wContribFromLibrary[25] * wSizeOfMovement))
-
(int) (Math.cos (18*Math.PI/180) * (wContribFromLibrary[26] * wSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (wContribFromLibrary[27] * wSizeOfMovement))
-
(int) (Math.cos (42*Math.PI/180) * (wContribFromLibrary[28] * wSizeOfMovement))
-
(int) (Math.cos (54*Math.PI/180) * (wContribFromLibrary[29] * wSizeOfMovement))
-
(int) (Math.cos (66*Math.PI/180) * (wContribFromLibrary[30] * wSizeOfMovement))
-
(int) (Math.cos (78*Math.PI/180) * (wContribFromLibrary[31] * wSizeOfMovement));

yWContig=yCircle
- (int) (wContribFromLibrary[0] * wSizeOfMovement)
-
(int) (Math.sin (79*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
-
(int) (Math.sin (68*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
-
(int) (Math.sin (57*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
-
(int) (Math.sin (46*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
-
(int) (Math.sin (35*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))
-
(int) (Math.sin (24*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
-
(int) (Math.sin (13*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
-
(int) (Math.sin (2*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
+ (int) (Math.sin (9*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
+ (int) (Math.sin (20*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
+ (int) (Math.sin (31*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
+ (int) (Math.sin (53*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
+ (int) (Math.sin (64*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
+ (int) (Math.sin (75*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))
+ (int) (Math.sin (86*Math.PI/180) * (wContribFromLibrary[16] * wSizeOfMovement))
+ (int) (Math.sin (83*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
+ (int) (Math.sin (72*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
+ (int) (Math.sin (61*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
+ (int) (Math.sin (50*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))

```

```

+ (int) (Math.sin(39*Math.PI/180) * (wContribFromLibrary[21] * wSizeOfMovement))
+ (int) (Math.sin(28*Math.PI/180) * (wContribFromLibrary[22] * wSizeOfMovement))
+ (int) (Math.sin(17*Math.PI/180) * (wContribFromLibrary[23] * wSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (wContribFromLibrary[24] * wSizeOfMovement))
-
(int) (Math.sin(6*Math.PI/180) * (wContribFromLibrary[25] * wSizeOfMovement))
-
(int) (Math.sin(18*Math.PI/180) * (wContribFromLibrary[26] * wSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[27] * wSizeOfMovement))
-
(int) (Math.sin(42*Math.PI/180) * (wContribFromLibrary[28] * wSizeOfMovement))
-
(int) (Math.sin(54*Math.PI/180) * (wContribFromLibrary[29] * wSizeOfMovement))
-
(int) (Math.sin(66*Math.PI/180) * (wContribFromLibrary[30] * wSizeOfMovement))
-
(int) (Math.sin(78*Math.PI/180) * (wContribFromLibrary[31] * wSizeOfMovement));
} // end 32

// if 33 libraries, 11 & 10-degree pies
else if ( 33 == numberOfLibraries)
{
    xPContig=xCircle
+ (int) (Math.cos(79*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(68*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.cos(57*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (Math.cos(46*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
+ (int) (Math.cos(35*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
    + (int) (Math.cos(24*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
+ (int) (Math.cos(13*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
+ (int) (Math.cos(2*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))
    + (int) (Math.cos(9*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
    + (int) (Math.cos(20*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))
+ (int) (Math.cos(31*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement))
+ (int) (Math.cos(53*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
+ (int) (Math.cos(64*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))

```

```

+ (int) (Math.cos (75*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))
+ (int) (Math.cos (86*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))
-
(int) (Math.cos (83*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement))
-
(int) (Math.cos (72*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement))
-
(int) (Math.cos (61*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (pContribFromLibrary [20] *pSizeOfMovement))
-
(int) (Math.cos (39*Math.PI/180) * (pContribFromLibrary [21] *pSizeOfMovement))
-
(int) (Math.cos (28*Math.PI/180) * (pContribFromLibrary [22] *pSizeOfMovement))
-
(int) (Math.cos (17*Math.PI/180) * (pContribFromLibrary [23] *pSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (pContribFromLibrary [24] *pSizeOfMovement))
-
(int) (Math.cos (5*Math.PI/180) * (pContribFromLibrary [25] *pSizeOfMovement))
-
(int) (Math.cos (16*Math.PI/180) * (pContribFromLibrary [26] *pSizeOfMovement))
-
(int) (Math.cos (27*Math.PI/180) * (pContribFromLibrary [27] *pSizeOfMovement))
-
(int) (Math.cos (38*Math.PI/180) * (pContribFromLibrary [28] *pSizeOfMovement))
-
(int) (Math.cos (49*Math.PI/180) * (pContribFromLibrary [29] *pSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (pContribFromLibrary [30] *pSizeOfMovement))
-
(int) (Math.cos (70*Math.PI/180) * (pContribFromLibrary [31] *pSizeOfMovement))
-
(int) (Math.cos (80*Math.PI/180) * (pContribFromLibrary [32] *pSizeOfMovement));

yPContig=yCircle
- (int) (pContribFromLibrary [0] *pSizeOfMovement)
-
(int) (Math.sin (79*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))
-
(int) (Math.sin (68*Math.PI/180) * (pContribFromLibrary [2] *pSizeOfMovement))
-
(int) (Math.sin (57*Math.PI/180) * (pContribFromLibrary [3] *pSizeOfMovement))
-
(int) (Math.sin (46*Math.PI/180) * (pContribFromLibrary [4] *pSizeOfMovement))
-
(int) (Math.sin (35*Math.PI/180) * (pContribFromLibrary [5] *pSizeOfMovement))
-
(int) (Math.sin (24*Math.PI/180) * (pContribFromLibrary [6] *pSizeOfMovement))
-
(int) (Math.sin (13*Math.PI/180) * (pContribFromLibrary [7] *pSizeOfMovement))
-
(int) (Math.sin (2*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
+ (int) (Math.sin (9*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))

```

```

+(int) (Math.sin(20*Math.PI/180) * (pContribFromLibrary[10] *pSizeOfMovement))
+(int) (Math.sin(31*Math.PI/180) * (pContribFromLibrary[11] *pSizeOfMovement))
+(int) (Math.sin(42*Math.PI/180) * (pContribFromLibrary[12] *pSizeOfMovement))
+(int) (Math.sin(53*Math.PI/180) * (pContribFromLibrary[13] *pSizeOfMovement))
+(int) (Math.sin(64*Math.PI/180) * (pContribFromLibrary[14] *pSizeOfMovement))
+(int) (Math.sin(75*Math.PI/180) * (pContribFromLibrary[15] *pSizeOfMovement))
+(int) (Math.sin(86*Math.PI/180) * (pContribFromLibrary[16] *pSizeOfMovement))
+(int) (Math.sin(83*Math.PI/180) * (pContribFromLibrary[17] *pSizeOfMovement))
+(int) (Math.sin(72*Math.PI/180) * (pContribFromLibrary[18] *pSizeOfMovement))
+(int) (Math.sin(61*Math.PI/180) * (pContribFromLibrary[19] *pSizeOfMovement))
+(int) (Math.sin(50*Math.PI/180) * (pContribFromLibrary[20] *pSizeOfMovement))
+(int) (Math.sin(39*Math.PI/180) * (pContribFromLibrary[21] *pSizeOfMovement))
+(int) (Math.sin(28*Math.PI/180) * (pContribFromLibrary[22] *pSizeOfMovement))
+(int) (Math.sin(17*Math.PI/180) * (pContribFromLibrary[23] *pSizeOfMovement))
+(int) (Math.sin(6*Math.PI/180) * (pContribFromLibrary[24] *pSizeOfMovement))
-
(int) (Math.sin(5*Math.PI/180) * (pContribFromLibrary[25] *pSizeOfMovement))
-
(int) (Math.sin(16*Math.PI/180) * (pContribFromLibrary[26] *pSizeOfMovement))
-
(int) (Math.sin(27*Math.PI/180) * (pContribFromLibrary[27] *pSizeOfMovement))
-
(int) (Math.sin(38*Math.PI/180) * (pContribFromLibrary[28] *pSizeOfMovement))
-
(int) (Math.sin(49*Math.PI/180) * (pContribFromLibrary[29] *pSizeOfMovement))
-
(int) (Math.sin(60*Math.PI/180) * (pContribFromLibrary[30] *pSizeOfMovement))
-
(int) (Math.sin(70*Math.PI/180) * (pContribFromLibrary[31] *pSizeOfMovement))
-
(int) (Math.sin(80*Math.PI/180) * (pContribFromLibrary[32] *pSizeOfMovement));;

```

xEContig=xCircle

```

+(int) (Math.cos(79*Math.PI/180) * (eContribFromLibrary[1] *eSizeOfMovement))
+(int) (Math.cos(68*Math.PI/180) * (eContribFromLibrary[2] *eSizeOfMovement))
+(int) (Math.cos(57*Math.PI/180) * (eContribFromLibrary[3] *eSizeOfMovement))
+(int) (Math.cos(46*Math.PI/180) * (eContribFromLibrary[4] *eSizeOfMovement))

```

```

+ (int) (Math.cos (35*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))
      + (int) (Math.cos (24*Math.PI/180) * (eContribFromLibrary [6] *eSizeOfMovement))
+ (int) (Math.cos (13*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
+ (int) (Math.cos (2*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
      + (int) (Math.cos (9*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))
      + (int) (Math.cos (20*Math.PI/180) * (eContribFromLibrary [10] *eSizeOfMovement))

      + (int) (Math.cos (31*Math.PI/180) * (eContribFromLibrary [11] *eSizeOfMovement))
+ (int) (Math.cos (42*Math.PI/180) * (eContribFromLibrary [12] *eSizeOfMovement))
+ (int) (Math.cos (53*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement))
+ (int) (Math.cos (64*Math.PI/180) * (eContribFromLibrary [14] *eSizeOfMovement))
+ (int) (Math.cos (75*Math.PI/180) * (eContribFromLibrary [15] *eSizeOfMovement))
+ (int) (Math.cos (86*Math.PI/180) * (eContribFromLibrary [16] *eSizeOfMovement))
-
(int) (Math.cos (83*Math.PI/180) * (eContribFromLibrary [17] *eSizeOfMovement))
-
(int) (Math.cos (72*Math.PI/180) * (eContribFromLibrary [18] *eSizeOfMovement))
-
(int) (Math.cos (61*Math.PI/180) * (eContribFromLibrary [19] *eSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (eContribFromLibrary [20] *eSizeOfMovement))
-
(int) (Math.cos (39*Math.PI/180) * (eContribFromLibrary [21] *eSizeOfMovement))
-
(int) (Math.cos (28*Math.PI/180) * (eContribFromLibrary [22] *eSizeOfMovement))
-
(int) (Math.cos (17*Math.PI/180) * (eContribFromLibrary [23] *eSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (eContribFromLibrary [24] *eSizeOfMovement))
-
(int) (Math.cos (5*Math.PI/180) * (eContribFromLibrary [25] *eSizeOfMovement))
-
(int) (Math.cos (16*Math.PI/180) * (eContribFromLibrary [26] *eSizeOfMovement))
-
(int) (Math.cos (27*Math.PI/180) * (eContribFromLibrary [27] *eSizeOfMovement))
-
(int) (Math.cos (38*Math.PI/180) * (eContribFromLibrary [28] *eSizeOfMovement))
-
(int) (Math.cos (49*Math.PI/180) * (eContribFromLibrary [29] *eSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (eContribFromLibrary [30] *eSizeOfMovement))
-
(int) (Math.cos (70*Math.PI/180) * (eContribFromLibrary [31] *eSizeOfMovement))
-
(int) (Math.cos (80*Math.PI/180) * (eContribFromLibrary [32] *eSizeOfMovement));

```

```

yEContig=yCircle
      - (int) (eContribFromLibrary[0] *eSizeOfMovement)
      -
(int) (Math.sin(79*Math.PI/180) * (eContribFromLibrary[1] *eSizeOfMovement))
      -
(int) (Math.sin(68*Math.PI/180) * (eContribFromLibrary[2] *eSizeOfMovement))
      -
(int) (Math.sin(57*Math.PI/180) * (eContribFromLibrary[3] *eSizeOfMovement))
      -
(int) (Math.sin(46*Math.PI/180) * (eContribFromLibrary[4] *eSizeOfMovement))
      -
(int) (Math.sin(35*Math.PI/180) * (eContribFromLibrary[5] *eSizeOfMovement))
      -
(int) (Math.sin(24*Math.PI/180) * (eContribFromLibrary[6] *eSizeOfMovement))
      -
(int) (Math.sin(13*Math.PI/180) * (eContribFromLibrary[7] *eSizeOfMovement))
      -
(int) (Math.sin(2*Math.PI/180) * (eContribFromLibrary[8] *eSizeOfMovement))
+ (int) (Math.sin(9*Math.PI/180) * (eContribFromLibrary[9] *eSizeOfMovement))
+ (int) (Math.sin(20*Math.PI/180) * (eContribFromLibrary[10] *eSizeOfMovement))
+ (int) (Math.sin(31*Math.PI/180) * (eContribFromLibrary[11] *eSizeOfMovement))
+ (int) (Math.sin(42*Math.PI/180) * (eContribFromLibrary[12] *eSizeOfMovement))
+ (int) (Math.sin(53*Math.PI/180) * (eContribFromLibrary[13] *eSizeOfMovement))
+ (int) (Math.sin(64*Math.PI/180) * (eContribFromLibrary[14] *eSizeOfMovement))
+ (int) (Math.sin(75*Math.PI/180) * (eContribFromLibrary[15] *eSizeOfMovement))
+ (int) (Math.sin(86*Math.PI/180) * (eContribFromLibrary[16] *eSizeOfMovement))
+ (int) (Math.sin(83*Math.PI/180) * (eContribFromLibrary[17] *eSizeOfMovement))
+ (int) (Math.sin(72*Math.PI/180) * (eContribFromLibrary[18] *eSizeOfMovement))
+ (int) (Math.sin(61*Math.PI/180) * (eContribFromLibrary[19] *eSizeOfMovement))
+ (int) (Math.sin(50*Math.PI/180) * (eContribFromLibrary[20] *eSizeOfMovement))
+ (int) (Math.sin(39*Math.PI/180) * (eContribFromLibrary[21] *eSizeOfMovement))
+ (int) (Math.sin(28*Math.PI/180) * (eContribFromLibrary[22] *eSizeOfMovement))
+ (int) (Math.sin(17*Math.PI/180) * (eContribFromLibrary[23] *eSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (eContribFromLibrary[24] *eSizeOfMovement))
      -
(int) (Math.sin(5*Math.PI/180) * (eContribFromLibrary[25] *eSizeOfMovement))
      -
(int) (Math.sin(16*Math.PI/180) * (eContribFromLibrary[26] *eSizeOfMovement))
      -
(int) (Math.sin(27*Math.PI/180) * (eContribFromLibrary[27] *eSizeOfMovement))

```

```

-
(int) (Math.sin(38*Math.PI/180) * (eContribFromLibrary[28] * eSizeOfMovement))
-
(int) (Math.sin(49*Math.PI/180) * (eContribFromLibrary[29] * eSizeOfMovement))
-
(int) (Math.sin(60*Math.PI/180) * (eContribFromLibrary[30] * eSizeOfMovement))
-
(int) (Math.sin(70*Math.PI/180) * (eContribFromLibrary[31] * eSizeOfMovement))
-
(int) (Math.sin(80*Math.PI/180) * (eContribFromLibrary[32] * eSizeOfMovement));

    xWContig=xCircle

+ (int) (Math.cos(79*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(68*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(57*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(46*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(35*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))

    + (int) (Math.cos(24*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.cos(13*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.cos(2*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))

    + (int) (Math.cos(9*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
    + (int) (Math.cos(20*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))

    + (int) (Math.cos(31*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
+ (int) (Math.cos(53*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
+ (int) (Math.cos(64*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
+ (int) (Math.cos(75*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))
+ (int) (Math.cos(86*Math.PI/180) * (wContribFromLibrary[16] * wSizeOfMovement))

-
(int) (Math.cos(83*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
-
(int) (Math.cos(72*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
-
(int) (Math.cos(61*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
-
(int) (Math.cos(50*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))
-
(int) (Math.cos(39*Math.PI/180) * (wContribFromLibrary[21] * wSizeOfMovement))
-
(int) (Math.cos(28*Math.PI/180) * (wContribFromLibrary[22] * wSizeOfMovement))

```

```

-
(int) (Math.cos (17*Math.PI/180) * (wContribFromLibrary [23] *wSizeOfMovement))
-
(int) (Math.cos (6*Math.PI/180) * (wContribFromLibrary [24] *wSizeOfMovement))
-
(int) (Math.cos (5*Math.PI/180) * (wContribFromLibrary [25] *wSizeOfMovement))
-
(int) (Math.cos (16*Math.PI/180) * (wContribFromLibrary [26] *wSizeOfMovement))
-
(int) (Math.cos (27*Math.PI/180) * (wContribFromLibrary [27] *wSizeOfMovement))
-
(int) (Math.cos (38*Math.PI/180) * (wContribFromLibrary [28] *wSizeOfMovement))
-
(int) (Math.cos (49*Math.PI/180) * (wContribFromLibrary [29] *wSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (wContribFromLibrary [30] *wSizeOfMovement))
-
(int) (Math.cos (70*Math.PI/180) * (wContribFromLibrary [31] *wSizeOfMovement))
-
(int) (Math.cos (80*Math.PI/180) * (wContribFromLibrary [32] *wSizeOfMovement));

    yWContig=yCircle
        - (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (79*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
-
(int) (Math.sin (68*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
-
(int) (Math.sin (57*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
-
(int) (Math.sin (46*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
-
(int) (Math.sin (35*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
-
(int) (Math.sin (24*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))
-
(int) (Math.sin (13*Math.PI/180) * (wContribFromLibrary [7] *wSizeOfMovement))
-
(int) (Math.sin (2*Math.PI/180) * (wContribFromLibrary [8] *wSizeOfMovement))
+ (int) (Math.sin (9*Math.PI/180) * (wContribFromLibrary [9] *wSizeOfMovement))
+ (int) (Math.sin (20*Math.PI/180) * (wContribFromLibrary [10] *wSizeOfMovement))
+ (int) (Math.sin (31*Math.PI/180) * (wContribFromLibrary [11] *wSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (wContribFromLibrary [12] *wSizeOfMovement))
+ (int) (Math.sin (53*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
+ (int) (Math.sin (64*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
+ (int) (Math.sin (75*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
+ (int) (Math.sin (86*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))
+ (int) (Math.sin (83*Math.PI/180) * (wContribFromLibrary [17] *wSizeOfMovement))

```

```

+ (int) (Math.sin(72*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
+ (int) (Math.sin(61*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
+ (int) (Math.sin(50*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))
+ (int) (Math.sin(39*Math.PI/180) * (wContribFromLibrary[21] * wSizeOfMovement))
+ (int) (Math.sin(28*Math.PI/180) * (wContribFromLibrary[22] * wSizeOfMovement))
+ (int) (Math.sin(17*Math.PI/180) * (wContribFromLibrary[23] * wSizeOfMovement))
+ (int) (Math.sin(6*Math.PI/180) * (wContribFromLibrary[24] * wSizeOfMovement))
-
(int) (Math.sin(5*Math.PI/180) * (wContribFromLibrary[25] * wSizeOfMovement))
-
(int) (Math.sin(16*Math.PI/180) * (wContribFromLibrary[26] * wSizeOfMovement))
-
(int) (Math.sin(27*Math.PI/180) * (wContribFromLibrary[27] * wSizeOfMovement))
-
(int) (Math.sin(38*Math.PI/180) * (wContribFromLibrary[28] * wSizeOfMovement))
-
(int) (Math.sin(49*Math.PI/180) * (wContribFromLibrary[29] * wSizeOfMovement))
-
(int) (Math.sin(60*Math.PI/180) * (wContribFromLibrary[30] * wSizeOfMovement))
-
(int) (Math.sin(70*Math.PI/180) * (wContribFromLibrary[31] * wSizeOfMovement))
-
(int) (Math.sin(80*Math.PI/180) * (wContribFromLibrary[32] * wSizeOfMovement));
    } // end 33

} // end calcXY_31to33()
//*****

void calcXY_34()
//*****

{
    // if 34 libraries, 11 & 10-degree pies
    if ( 34 == numberOfLibraries)
    {
        xPContig=xCircle

+ (int) (Math.cos(79*Math.PI/180) * (pContribFromLibrary[1] * pSizeOfMovement))
+ (int) (Math.cos(68*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
+ (int) (Math.cos(57*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
+ (int) (Math.cos(46*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
+ (int) (Math.cos(35*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))

        + (int) (Math.cos(24*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))

+ (int) (Math.cos(13*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))

```

```

+ (int) (Math.cos (2*Math.PI/180) * (pContribFromLibrary [8] *pSizeOfMovement))
    + (int) (Math.cos (9*Math.PI/180) * (pContribFromLibrary [9] *pSizeOfMovement))
    + (int) (Math.cos (20*Math.PI/180) * (pContribFromLibrary [10] *pSizeOfMovement))
+ (int) (Math.cos (31*Math.PI/180) * (pContribFromLibrary [11] *pSizeOfMovement))
+ (int) (Math.cos (42*Math.PI/180) * (pContribFromLibrary [12] *pSizeOfMovement))
+ (int) (Math.cos (53*Math.PI/180) * (pContribFromLibrary [13] *pSizeOfMovement))
+ (int) (Math.cos (64*Math.PI/180) * (pContribFromLibrary [14] *pSizeOfMovement))
+ (int) (Math.cos (75*Math.PI/180) * (pContribFromLibrary [15] *pSizeOfMovement))
+ (int) (Math.cos (86*Math.PI/180) * (pContribFromLibrary [16] *pSizeOfMovement))
-
(int) (Math.cos (83*Math.PI/180) * (pContribFromLibrary [17] *pSizeOfMovement))
-
(int) (Math.cos (72*Math.PI/180) * (pContribFromLibrary [18] *pSizeOfMovement))
-
(int) (Math.cos (61*Math.PI/180) * (pContribFromLibrary [19] *pSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (pContribFromLibrary [20] *pSizeOfMovement))
-
(int) (Math.cos (40*Math.PI/180) * (pContribFromLibrary [21] *pSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [22] *pSizeOfMovement))
-
(int) (Math.cos (20*Math.PI/180) * (pContribFromLibrary [23] *pSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (pContribFromLibrary [24] *pSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (pContribFromLibrary [26] *pSizeOfMovement))
-
(int) (Math.cos (20*Math.PI/180) * (pContribFromLibrary [27] *pSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (pContribFromLibrary [28] *pSizeOfMovement))
-
(int) (Math.cos (40*Math.PI/180) * (pContribFromLibrary [29] *pSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (pContribFromLibrary [30] *pSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (pContribFromLibrary [31] *pSizeOfMovement))
-
(int) (Math.cos (70*Math.PI/180) * (pContribFromLibrary [32] *pSizeOfMovement))
-
(int) (Math.cos (80*Math.PI/180) * (pContribFromLibrary [33] *pSizeOfMovement));

yPContig=yCircle
    - (int) (pContribFromLibrary [0] *pSizeOfMovement)
    -
(int) (Math.sin (79*Math.PI/180) * (pContribFromLibrary [1] *pSizeOfMovement))

```

```

-
(int) (Math.sin(68*Math.PI/180) * (pContribFromLibrary[2] * pSizeOfMovement))
-
(int) (Math.sin(57*Math.PI/180) * (pContribFromLibrary[3] * pSizeOfMovement))
-
(int) (Math.sin(46*Math.PI/180) * (pContribFromLibrary[4] * pSizeOfMovement))
-
(int) (Math.sin(35*Math.PI/180) * (pContribFromLibrary[5] * pSizeOfMovement))
-
(int) (Math.sin(24*Math.PI/180) * (pContribFromLibrary[6] * pSizeOfMovement))
-
(int) (Math.sin(13*Math.PI/180) * (pContribFromLibrary[7] * pSizeOfMovement))
-
(int) (Math.sin(2*Math.PI/180) * (pContribFromLibrary[8] * pSizeOfMovement))
+ (int) (Math.sin(9*Math.PI/180) * (pContribFromLibrary[9] * pSizeOfMovement))
+ (int) (Math.sin(20*Math.PI/180) * (pContribFromLibrary[10] * pSizeOfMovement))
+ (int) (Math.sin(31*Math.PI/180) * (pContribFromLibrary[11] * pSizeOfMovement))
+ (int) (Math.sin(42*Math.PI/180) * (pContribFromLibrary[12] * pSizeOfMovement))
+ (int) (Math.sin(53*Math.PI/180) * (pContribFromLibrary[13] * pSizeOfMovement))
+ (int) (Math.sin(64*Math.PI/180) * (pContribFromLibrary[14] * pSizeOfMovement))
+ (int) (Math.sin(75*Math.PI/180) * (pContribFromLibrary[15] * pSizeOfMovement))
+ (int) (Math.sin(86*Math.PI/180) * (pContribFromLibrary[16] * pSizeOfMovement))
+ (int) (Math.sin(83*Math.PI/180) * (pContribFromLibrary[17] * pSizeOfMovement))
+ (int) (Math.sin(72*Math.PI/180) * (pContribFromLibrary[18] * pSizeOfMovement))
+ (int) (Math.sin(61*Math.PI/180) * (pContribFromLibrary[19] * pSizeOfMovement))
+ (int) (Math.sin(50*Math.PI/180) * (pContribFromLibrary[20] * pSizeOfMovement))
+ (int) (Math.sin(40*Math.PI/180) * (pContribFromLibrary[21] * pSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[22] * pSizeOfMovement))
+ (int) (Math.sin(20*Math.PI/180) * (pContribFromLibrary[23] * pSizeOfMovement))
+ (int) (Math.sin(10*Math.PI/180) * (pContribFromLibrary[24] * pSizeOfMovement))

-
(int) (Math.sin(10*Math.PI/180) * (pContribFromLibrary[26] * pSizeOfMovement))
-
(int) (Math.sin(20*Math.PI/180) * (pContribFromLibrary[27] * pSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (pContribFromLibrary[28] * pSizeOfMovement))
-
(int) (Math.sin(40*Math.PI/180) * (pContribFromLibrary[29] * pSizeOfMovement))
-
(int) (Math.sin(50*Math.PI/180) * (pContribFromLibrary[30] * pSizeOfMovement))

```

```

-
(int) (Math.sin(60*Math.PI/180) * (pContribFromLibrary[31] * pSizeOfMovement))
-
(int) (Math.sin(70*Math.PI/180) * (pContribFromLibrary[32] * pSizeOfMovement))
-
(int) (Math.sin(80*Math.PI/180) * (pContribFromLibrary[33] * pSizeOfMovement));

    xEContig=xCircle

+ (int) (Math.cos(79*Math.PI/180) * (eContribFromLibrary[1] * eSizeOfMovement))
+ (int) (Math.cos(68*Math.PI/180) * (eContribFromLibrary[2] * eSizeOfMovement))
+ (int) (Math.cos(57*Math.PI/180) * (eContribFromLibrary[3] * eSizeOfMovement))
+ (int) (Math.cos(46*Math.PI/180) * (eContribFromLibrary[4] * eSizeOfMovement))
+ (int) (Math.cos(35*Math.PI/180) * (eContribFromLibrary[5] * eSizeOfMovement))

    + (int) (Math.cos(24*Math.PI/180) * (eContribFromLibrary[6] * eSizeOfMovement))
+ (int) (Math.cos(13*Math.PI/180) * (eContribFromLibrary[7] * eSizeOfMovement))
+ (int) (Math.cos(2*Math.PI/180) * (eContribFromLibrary[8] * eSizeOfMovement))

    + (int) (Math.cos(9*Math.PI/180) * (eContribFromLibrary[9] * eSizeOfMovement))
    + (int) (Math.cos(20*Math.PI/180) * (eContribFromLibrary[10] * eSizeOfMovement))

    + (int) (Math.cos(31*Math.PI/180) * (eContribFromLibrary[11] * eSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (eContribFromLibrary[12] * eSizeOfMovement))
+ (int) (Math.cos(53*Math.PI/180) * (eContribFromLibrary[13] * eSizeOfMovement))
+ (int) (Math.cos(64*Math.PI/180) * (eContribFromLibrary[14] * eSizeOfMovement))
+ (int) (Math.cos(75*Math.PI/180) * (eContribFromLibrary[15] * eSizeOfMovement))
+ (int) (Math.cos(86*Math.PI/180) * (eContribFromLibrary[16] * eSizeOfMovement))

-
(int) (Math.cos(83*Math.PI/180) * (eContribFromLibrary[17] * eSizeOfMovement))
-
(int) (Math.cos(72*Math.PI/180) * (eContribFromLibrary[18] * eSizeOfMovement))
-
(int) (Math.cos(61*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
-
(int) (Math.cos(50*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
-
(int) (Math.cos(40*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement))
-
(int) (Math.cos(30*Math.PI/180) * (eContribFromLibrary[22] * eSizeOfMovement))
-
(int) (Math.cos(20*Math.PI/180) * (eContribFromLibrary[23] * eSizeOfMovement))
-
(int) (Math.cos(10*Math.PI/180) * (eContribFromLibrary[24] * eSizeOfMovement))

```

```

-
(int) (Math.cos (10*Math.PI/180) * (eContribFromLibrary [26] *eSizeOfMovement))
-
(int) (Math.cos (20*Math.PI/180) * (eContribFromLibrary [27] *eSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (eContribFromLibrary [28] *eSizeOfMovement))
-
(int) (Math.cos (40*Math.PI/180) * (eContribFromLibrary [29] *eSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (eContribFromLibrary [30] *eSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (eContribFromLibrary [31] *eSizeOfMovement))
-
(int) (Math.cos (70*Math.PI/180) * (eContribFromLibrary [32] *eSizeOfMovement))
-
(int) (Math.cos (80*Math.PI/180) * (eContribFromLibrary [33] *eSizeOfMovement));

yEContig=yCircle
- (int) (eContribFromLibrary [0] *eSizeOfMovement)
-
(int) (Math.sin (79*Math.PI/180) * (eContribFromLibrary [1] *eSizeOfMovement))
-
(int) (Math.sin (68*Math.PI/180) * (eContribFromLibrary [2] *eSizeOfMovement))
-
(int) (Math.sin (57*Math.PI/180) * (eContribFromLibrary [3] *eSizeOfMovement))
-
(int) (Math.sin (46*Math.PI/180) * (eContribFromLibrary [4] *eSizeOfMovement))
-
(int) (Math.sin (35*Math.PI/180) * (eContribFromLibrary [5] *eSizeOfMovement))
-
(int) (Math.sin (24*Math.PI/180) * (eContribFromLibrary [6] *eSizeOfMovement))
-
(int) (Math.sin (13*Math.PI/180) * (eContribFromLibrary [7] *eSizeOfMovement))
-
(int) (Math.sin (2*Math.PI/180) * (eContribFromLibrary [8] *eSizeOfMovement))
+ (int) (Math.sin (9*Math.PI/180) * (eContribFromLibrary [9] *eSizeOfMovement))
+ (int) (Math.sin (20*Math.PI/180) * (eContribFromLibrary [10] *eSizeOfMovement))
+ (int) (Math.sin (31*Math.PI/180) * (eContribFromLibrary [11] *eSizeOfMovement))
+ (int) (Math.sin (42*Math.PI/180) * (eContribFromLibrary [12] *eSizeOfMovement))
+ (int) (Math.sin (53*Math.PI/180) * (eContribFromLibrary [13] *eSizeOfMovement))
+ (int) (Math.sin (64*Math.PI/180) * (eContribFromLibrary [14] *eSizeOfMovement))
+ (int) (Math.sin (75*Math.PI/180) * (eContribFromLibrary [15] *eSizeOfMovement))
+ (int) (Math.sin (86*Math.PI/180) * (eContribFromLibrary [16] *eSizeOfMovement))
+ (int) (Math.sin (83*Math.PI/180) * (eContribFromLibrary [17] *eSizeOfMovement))
+ (int) (Math.sin (72*Math.PI/180) * (eContribFromLibrary [18] *eSizeOfMovement))

```

```

+ (int) (Math.sin(61*Math.PI/180) * (eContribFromLibrary[19] * eSizeOfMovement))
+ (int) (Math.sin(50*Math.PI/180) * (eContribFromLibrary[20] * eSizeOfMovement))
+ (int) (Math.sin(40*Math.PI/180) * (eContribFromLibrary[21] * eSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[22] * eSizeOfMovement))
+ (int) (Math.sin(20*Math.PI/180) * (eContribFromLibrary[23] * eSizeOfMovement))
+ (int) (Math.sin(10*Math.PI/180) * (eContribFromLibrary[24] * eSizeOfMovement))

```

```

-
(int) (Math.sin(10*Math.PI/180) * (eContribFromLibrary[26] * eSizeOfMovement))
-
(int) (Math.sin(20*Math.PI/180) * (eContribFromLibrary[27] * eSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (eContribFromLibrary[28] * eSizeOfMovement))
-
(int) (Math.sin(40*Math.PI/180) * (eContribFromLibrary[29] * eSizeOfMovement))
-
(int) (Math.sin(50*Math.PI/180) * (eContribFromLibrary[30] * eSizeOfMovement))
-
(int) (Math.sin(60*Math.PI/180) * (eContribFromLibrary[31] * eSizeOfMovement))
-
(int) (Math.sin(70*Math.PI/180) * (eContribFromLibrary[32] * eSizeOfMovement))
-
(int) (Math.sin(80*Math.PI/180) * (eContribFromLibrary[33] * eSizeOfMovement));

```

 xWContig=xCircle

```

+ (int) (Math.cos(79*Math.PI/180) * (wContribFromLibrary[1] * wSizeOfMovement))
+ (int) (Math.cos(68*Math.PI/180) * (wContribFromLibrary[2] * wSizeOfMovement))
+ (int) (Math.cos(57*Math.PI/180) * (wContribFromLibrary[3] * wSizeOfMovement))
+ (int) (Math.cos(46*Math.PI/180) * (wContribFromLibrary[4] * wSizeOfMovement))
+ (int) (Math.cos(35*Math.PI/180) * (wContribFromLibrary[5] * wSizeOfMovement))

+ (int) (Math.cos(24*Math.PI/180) * (wContribFromLibrary[6] * wSizeOfMovement))
+ (int) (Math.cos(13*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
+ (int) (Math.cos(2*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))

+ (int) (Math.cos(9*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
+ (int) (Math.cos(20*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))

+ (int) (Math.cos(31*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
+ (int) (Math.cos(42*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))

```

```

+ (int) (Math.cos (53*Math.PI/180) * (wContribFromLibrary [13] *wSizeOfMovement))
+ (int) (Math.cos (64*Math.PI/180) * (wContribFromLibrary [14] *wSizeOfMovement))
+ (int) (Math.cos (75*Math.PI/180) * (wContribFromLibrary [15] *wSizeOfMovement))
+ (int) (Math.cos (86*Math.PI/180) * (wContribFromLibrary [16] *wSizeOfMovement))
-
(int) (Math.cos (83*Math.PI/180) * (wContribFromLibrary [17] *wSizeOfMovement))
-
(int) (Math.cos (72*Math.PI/180) * (wContribFromLibrary [18] *wSizeOfMovement))
-
(int) (Math.cos (61*Math.PI/180) * (wContribFromLibrary [19] *wSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (wContribFromLibrary [20] *wSizeOfMovement))
-
(int) (Math.cos (40*Math.PI/180) * (wContribFromLibrary [21] *wSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (wContribFromLibrary [22] *wSizeOfMovement))
-
(int) (Math.cos (20*Math.PI/180) * (wContribFromLibrary [23] *wSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (wContribFromLibrary [24] *wSizeOfMovement))
-
(int) (Math.cos (10*Math.PI/180) * (wContribFromLibrary [26] *wSizeOfMovement))
-
(int) (Math.cos (20*Math.PI/180) * (wContribFromLibrary [27] *wSizeOfMovement))
-
(int) (Math.cos (30*Math.PI/180) * (wContribFromLibrary [28] *wSizeOfMovement))
-
(int) (Math.cos (40*Math.PI/180) * (wContribFromLibrary [29] *wSizeOfMovement))
-
(int) (Math.cos (50*Math.PI/180) * (wContribFromLibrary [30] *wSizeOfMovement))
-
(int) (Math.cos (60*Math.PI/180) * (wContribFromLibrary [31] *wSizeOfMovement))
-
(int) (Math.cos (70*Math.PI/180) * (wContribFromLibrary [32] *wSizeOfMovement))
-
(int) (Math.cos (80*Math.PI/180) * (wContribFromLibrary [33] *wSizeOfMovement));

yWContig=yCircle
- (int) (wContribFromLibrary [0] *wSizeOfMovement)
-
(int) (Math.sin (79*Math.PI/180) * (wContribFromLibrary [1] *wSizeOfMovement))
-
(int) (Math.sin (68*Math.PI/180) * (wContribFromLibrary [2] *wSizeOfMovement))
-
(int) (Math.sin (57*Math.PI/180) * (wContribFromLibrary [3] *wSizeOfMovement))
-
(int) (Math.sin (46*Math.PI/180) * (wContribFromLibrary [4] *wSizeOfMovement))
-
(int) (Math.sin (35*Math.PI/180) * (wContribFromLibrary [5] *wSizeOfMovement))
-
(int) (Math.sin (24*Math.PI/180) * (wContribFromLibrary [6] *wSizeOfMovement))

```

```

-
(int) (Math.sin(13*Math.PI/180) * (wContribFromLibrary[7] * wSizeOfMovement))
-
(int) (Math.sin(2*Math.PI/180) * (wContribFromLibrary[8] * wSizeOfMovement))
+ (int) (Math.sin(9*Math.PI/180) * (wContribFromLibrary[9] * wSizeOfMovement))
+ (int) (Math.sin(20*Math.PI/180) * (wContribFromLibrary[10] * wSizeOfMovement))
+ (int) (Math.sin(31*Math.PI/180) * (wContribFromLibrary[11] * wSizeOfMovement))
+ (int) (Math.sin(42*Math.PI/180) * (wContribFromLibrary[12] * wSizeOfMovement))
+ (int) (Math.sin(53*Math.PI/180) * (wContribFromLibrary[13] * wSizeOfMovement))
+ (int) (Math.sin(64*Math.PI/180) * (wContribFromLibrary[14] * wSizeOfMovement))
+ (int) (Math.sin(75*Math.PI/180) * (wContribFromLibrary[15] * wSizeOfMovement))
+ (int) (Math.sin(86*Math.PI/180) * (wContribFromLibrary[16] * wSizeOfMovement))
+ (int) (Math.sin(83*Math.PI/180) * (wContribFromLibrary[17] * wSizeOfMovement))
+ (int) (Math.sin(72*Math.PI/180) * (wContribFromLibrary[18] * wSizeOfMovement))
+ (int) (Math.sin(61*Math.PI/180) * (wContribFromLibrary[19] * wSizeOfMovement))
+ (int) (Math.sin(50*Math.PI/180) * (wContribFromLibrary[20] * wSizeOfMovement))
+ (int) (Math.sin(40*Math.PI/180) * (wContribFromLibrary[21] * wSizeOfMovement))
+ (int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[22] * wSizeOfMovement))
+ (int) (Math.sin(20*Math.PI/180) * (wContribFromLibrary[23] * wSizeOfMovement))
+ (int) (Math.sin(10*Math.PI/180) * (wContribFromLibrary[24] * wSizeOfMovement))

-
(int) (Math.sin(10*Math.PI/180) * (wContribFromLibrary[26] * wSizeOfMovement))
-
(int) (Math.sin(20*Math.PI/180) * (wContribFromLibrary[27] * wSizeOfMovement))
-
(int) (Math.sin(30*Math.PI/180) * (wContribFromLibrary[28] * wSizeOfMovement))
-
(int) (Math.sin(40*Math.PI/180) * (wContribFromLibrary[29] * wSizeOfMovement))
-
(int) (Math.sin(50*Math.PI/180) * (wContribFromLibrary[30] * wSizeOfMovement))
-
(int) (Math.sin(60*Math.PI/180) * (wContribFromLibrary[31] * wSizeOfMovement))
-
(int) (Math.sin(70*Math.PI/180) * (wContribFromLibrary[32] * wSizeOfMovement))
-
(int) (Math.sin(80*Math.PI/180) * (wContribFromLibrary[33] * wSizeOfMovement));
    } // end 34

} // end calcXY_34()

```

```

// REMOVE COMMENT LATER
*/
//*****

void reinitializeContribs() // do before calculating each contig's
coordinates
//*****

{
    eLibsPresent=0;
    pNumESTsPresent = 0;
    wContigSize = 0;

    for (int i = 0; i < numberOfLibraries; i++)
    {
        pContribFromLibrary[i]=0;
        eContribFromLibrary[i]=0;
        wContribFromLibrary[i]=0;
    }

} // end reinitializeContribs()
//*****

String[] getContigsForLibrary(String libraryName) // called by
showContigsFromLibrary()
//*****

{
    // find contigs having ESTs from this
    library
    String[] contigStrings = null;

    try
    {
        // get array of contig names associated with library
        contigStrings = server.getContigsFromLibrary(libraryName);
    }
    catch (Exception e) {
        err1+="getContigsForLibrary() error: "+e;
    }

    return contigStrings;
} // end getContigsForLibrary()
//*****

/**
// PROBABLY DON'T NEED (duplicates whatsInContig), BUT KEEP FOR NOW
String[] getContigESTs(String contigName)
{
    String[] ContigESTStrings = null;

    try
    {
        // get array of EST names associated with contig
        ContigESTStrings = server.getESTNamesOfContig(contigName);
    }
}
//works 4/23/02
catch (Exception e) {

```

```

        errl+="getContigESTs error: "+e;
    }

    return ContigESTStrings;
} // end getContigESTs()
*/
//*****

/**
// PROBABLY DON'T NEED, BUT KEEP FOR NOW
String[] getLibraryESTs(String libraryName)
{
    String[] LibESTStrings = null;

    try
    {
        // get array of EST names associated with library
        LibESTStrings = server.getESTNamesOfLibrary(libraryName);
    }
//works 4/23/02
    catch (Exception e) {
        errl+="getLibraryESTs error: "+e;
    }

    return LibESTStrings;
} // end getLibraryESTs()
*/
//*****

/**
// PROBABLY DON'T NEED, BUT KEEP FOR NOW
String[] getCtgLibESTs(String contigName, String libraryName)
{
    String[] CtgLibESTStrings = null;

    try
    {
        // get array of EST names associated with contig & library
        CtgLibESTStrings = server.getESTNames(contigName,libraryName);
    }
    catch (Exception e) {
        errl+="getCtgLibESTs error: "+e;
    }

    return CtgLibESTStrings;
} //end getCtgLibESTs
*/
//*****

void drawLibraries(Graphics g) // draw libraries on perimeter of circle
//*****

{
    try

```

```

{
    g.setColor(Color.red); // DARK_RED doesn't stand out enough

    // start drawing red dots around circle
    if ( 1 == numberOfLibraries)
    {
        x[0]=xCircle;
        y[0]=START_Y;
        g.fillOval(x[0],y[0],4,4); // top point LibA

        labelX[0]=x[0]-x_ADJUSTMENT; // move library name out of circle
        labelY[0]=y[0]-y_ADJUSTMENT; // move up

    } // end if numberOfLibraries is 1

    else if ( 2 == numberOfLibraries)
    {
        x[0]=xCircle; // grab display coordinates
        y[0]=START_Y;
        g.fillOval(x[0],y[0],4,4); // top of circle

        labelX[0]=x[0]-x_ADJUSTMENT; // move library name out of circle
        labelY[0]=y[0]-y_ADJUSTMENT; // move up

        x[1]=xCircle;
        y[1]=START_Y+2*CIRCLE_RADIUS_INT;
        g.fillOval(x[1],y[1],4,4); // bottom of circle

        labelX[1]=x[1]-x_ADJUSTMENT; // move library name out of circle
        labelY[1]=y[1]+y_ADJUSTMENT; // move down

    } // end else if numberOfLibraries is 2

    else if ( 3 == numberOfLibraries)
    {
        x[0]=xCircle; // display coordinates
        y[0]=START_Y;
        g.fillOval(x[0],y[0],4,4); // top point Lib1

        labelX[0]=x[0]-x_ADJUSTMENT; // move library name out of circle
        labelY[0]=y[0]-y_ADJUSTMENT; // move up

        // For documentation on the following trigonometric calculation, see
        // calculateXY() above.

        x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
        y[1]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
        g.fillOval(x[1],y[1],4,4); // right point Lib2

        labelX[1]=x[1]+3; // move library name out of circle
        labelY[1]=y[1]+y_ADJUSTMENT; // move down

        x[2]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
        y[2]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
    }
}

```

```

g.fillOval(x[2],y[2],4,4); // left point Lib3

labelX[2]=x[2]-x_ADJUSTMENT; // move library name out of circle
labelY[2]=y[2]+y_ADJUSTMENT; // move down

} // end else if numberOfLibraries is 3

else if ( 4 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4); // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT; // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT; // move up

    x[1]=START_X+2*CIRCLE_RADIUS_INT;
    y[1]=yCircle;
    g.fillOval(x[1],y[1],4,4); // right

    labelX[1]=x[1]+3; // move library name out of circle
    labelY[1]=y[1]+y_ADJUSTMENT; // move down

    x[2]=xCircle;
    y[2]=START_Y+2*CIRCLE_RADIUS_INT;
    g.fillOval(x[2],y[2],4,4); // bottom of circle

    labelX[2]=x[2]-x_ADJUSTMENT; // move library name out of circle
    labelY[2]=y[2]+y_ADJUSTMENT; // move down

    x[3]=START_X;
    y[3]=yCircle;
    g.fillOval(x[3],y[3],4,4); // left

    labelX[3]=x[3]-x_ADJUSTMENT; // move library name out of circle
    labelY[3]=y[3]+y_ADJUSTMENT; // move down

} // end if numberOfLibraries is 4

else if ( 5 == numberOfLibraries)
{
    x[0]=xCircle; // grab display coordinates
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4); // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT; // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT; // move up

    x[1]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
    y[1]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
    g.fillOval(x[1],y[1],4,4); // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(54*Math.PI/180));

```

```

y[2]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[2],y[2],4,4); // 2nd quadrant

labelX[2]=x[2]+3;
labelY[2]=y[2]+y_ADJUSTMENT;

x[3]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[3]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[3],y[3],4,4); // 3rd quadrant

labelX[3]=x[3]-x_ADJUSTMENT; // move library name out of circle
labelY[3]=y[3]+y_ADJUSTMENT; // move down

x[4]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[4]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[4],y[4],4,4); // 4th quadrant

labelX[4]=x[4]-x_ADJUSTMENT; // move library name out of circle
labelY[4]=y[4]-y_ADJUSTMENT; // move up

} // end else if numberOfLibraries is 5

else if ( 6 == numberOfLibraries)
{
    x[0]=xCircle; // grab display coordinates
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4); // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT; // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT; // move up

    x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
    y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
    g.fillOval(x[1],y[1],4,4); // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
    y[2]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
    g.fillOval(x[2],y[2],4,4); // 2nd quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]+y_ADJUSTMENT;

    x[3]=xCircle;
    y[3]=START_Y+2*CIRCLE_RADIUS_INT;
    g.fillOval(x[3],y[3],4,4); // bottom of circle

    labelX[3]=x[3]-x_ADJUSTMENT; // move library name out of circle
    labelY[3]=y[3]+y_ADJUSTMENT; // move down

    x[4]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
    y[4]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
    g.fillOval(x[4],y[4],4,4); // 3rd quadrant

    labelX[4]=x[4]-x_ADJUSTMENT; // move library name out of circle

```

```

labelY[4]=y[4]+y_ADJUSTMENT;    // move down

x[5]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[5]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[5],y[5],4,4);    // 4th quadrant

labelX[5]=x[5]-x_ADJUSTMENT;    // move library name out of circle
labelY[5]=y[5]-y_ADJUSTMENT;    // move up

} // end 6

else if ( 7 == numberOfLibraries)
{
    x[0]=xCircle;                // grab display coordinates
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

    x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(39*Math.PI/180));
    y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(39*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);    // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(12*Math.PI/180));
    y[2]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(12*Math.PI/180));
    g.fillOval(x[2],y[2],4,4);    // 2nd quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]+y_ADJUSTMENT;

    x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(63*Math.PI/180));
    y[3]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(63*Math.PI/180));
    g.fillOval(x[3],y[3],4,4);    // 2nd quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]+y_ADJUSTMENT;

    x[4]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
    y[4]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
    g.fillOval(x[4],y[4],4,4);    // 3rd quadrant

    labelX[4]=x[4]-x_ADJUSTMENT;    // move library name out of circle
    labelY[4]=y[4]+y_ADJUSTMENT;    // move down

    x[5]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(15*Math.PI/180));
    y[5]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(15*Math.PI/180));
    g.fillOval(x[5],y[5],4,4);    // 3rd quadrant

    labelX[5]=x[5]-x_ADJUSTMENT;    // move library name out of circle
    labelY[5]=y[5]+y_ADJUSTMENT;    // move down

    x[6]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(36*Math.PI/180));

```

```

y[6]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(36*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 4th quadrant

labelX[6]=x[6]-x_ADJUSTMENT; // move library name out of circle
labelY[6]=y[6]-y_ADJUSTMENT; // move up

} // end 7

else if ( 8 == numberOfLibraries)
{
x[0]=xCircle; // grab display coordinates
y[0]=START_Y;
g.fillOval(x[0],y[0],4,4); // top of circle

labelX[0]=x[0]-x_ADJUSTMENT; // move library name out of circle
labelY[0]=y[0]-y_ADJUSTMENT; // move up

x[1]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
y[1]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
g.fillOval(x[1],y[1],4,4); // 1st quadrant

labelX[1]=x[1]+3;
labelY[1]=y[1]-y_ADJUSTMENT;

x[2]=START_X+2*CIRCLE_RADIUS_INT;
y[2]=yCircle;
g.fillOval(x[2],y[2],4,4); // right

labelX[2]=x[2]+3;
labelY[2]=y[2]+y_ADJUSTMENT;

x[3]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
y[3]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
g.fillOval(x[3],y[3],4,4); // 2nd quadrant

labelX[3]=x[3]+3;
labelY[3]=y[3]+y_ADJUSTMENT;

x[4]=xCircle;
y[4]=START_Y+2*CIRCLE_RADIUS_INT;
g.fillOval(x[4],y[4],4,4); // bottom of circle

labelX[4]=x[4]-x_ADJUSTMENT; // move library name out of circle
labelY[4]=y[4]+y_ADJUSTMENT; // move down

x[5]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
y[5]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
g.fillOval(x[5],y[5],4,4); // 3rd quadrant

labelX[5]=x[5]-x_ADJUSTMENT; // move library name out of circle
labelY[5]=y[5]+y_ADJUSTMENT; // move down

x[6]=START_X;
y[6]=yCircle;
g.fillOval(x[6],y[6],4,4); // left

labelX[6]=x[6]-x_ADJUSTMENT; // move library name out of circle

```

```

labelY[6]=y[6]+y_ADJUSTMENT;          // move down

x[7]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
y[7]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 4th quadrant

labelX[7]=x[7]-x_ADJUSTMENT;          // move library name out of circle
labelY[7]=y[7]-y_ADJUSTMENT; // move up

} // end 8

else if ( 9 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4); // top point

    labelX[0]=x[0]-x_ADJUSTMENT;          // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT; // move up

    x[1]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(50*Math.PI/180));
    y[1]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(50*Math.PI/180));
    g.fillOval(x[1],y[1],4,4); // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(10*Math.PI/180));
    y[2]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(10*Math.PI/180));
    g.fillOval(x[2],y[2],4,4); // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
    y[3]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
    g.fillOval(x[3],y[3],4,4); // 2nd quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]+y_ADJUSTMENT;

    x[4]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(70*Math.PI/180));
    y[4]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(70*Math.PI/180));
    g.fillOval(x[4],y[4],4,4); // 2nd quadrant

    labelX[4]=x[4]+3;
    labelY[4]=y[4]+y_ADJUSTMENT;

    x[5]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(70*Math.PI/180));
    y[5]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(70*Math.PI/180));
    g.fillOval(x[5],y[5],4,4); // 3rd quadrant

    labelX[5]=x[5]-x_ADJUSTMENT;          // move library name out of circle
    labelY[5]=y[5]+y_ADJUSTMENT;          // move down

    x[6]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
    y[6]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(30*Math.PI/180));

```

```

g.fillOval(x[6],y[6],4,4); // 3rd quadrant

labelX[6]=x[6]-x_ADJUSTMENT; // move library name out of circle
labelY[6]=y[6]+y_ADJUSTMENT; // move down

x[7]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(10*Math.PI/180));
y[7]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(10*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 4th quadrant

labelX[7]=x[7]-x_ADJUSTMENT; // move library name out of circle
labelY[7]=y[7]-y_ADJUSTMENT; // move up

x[8]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(50*Math.PI/180));
y[8]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(50*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 4th quadrant

labelX[8]=x[8]-x_ADJUSTMENT; // move library name out of circle
labelY[8]=y[8]-y_ADJUSTMENT; // move up

} // end 9

else if ( 10 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4); // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT; // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT; // move up

    x[1]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
    y[1]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
    g.fillOval(x[1],y[1],4,4); // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
    y[2]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
    g.fillOval(x[2],y[2],4,4); // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
    y[3]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
    g.fillOval(x[3],y[3],4,4); // 2nd quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]+y_ADJUSTMENT;

    x[4]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
    y[4]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
    g.fillOval(x[4],y[4],4,4); // 2nd quadrant

    labelX[4]=x[4]+3;
    labelY[4]=y[4]+y_ADJUSTMENT;

```

```

x[5]=xCircle;
y[5]=START_Y+2*CIRCLE_RADIUS_INT;
g.fillOval(x[5],y[5],4,4); // bottom of circle

labelX[5]=x[5]-x_ADJUSTMENT; // move library name out of circle
labelY[5]=y[5]+y_ADJUSTMENT; // move down

x[6]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[6]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 3rd quadrant

labelX[6]=x[6]-x_ADJUSTMENT; // move library name out of circle
labelY[6]=y[6]+y_ADJUSTMENT; // move down

x[7]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[7]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 3rd quadrant

labelX[7]=x[7]-x_ADJUSTMENT; // move library name out of circle
labelY[7]=y[7]+y_ADJUSTMENT; // move down

x[8]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[8]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 4th quadrant

labelX[8]=x[8]-x_ADJUSTMENT; // move library name out of circle
labelY[8]=y[8]-y_ADJUSTMENT; // move up

x[9]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[9]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 4th quadrant

labelX[9]=x[9]-x_ADJUSTMENT; // move library name out of circle
labelY[9]=y[9]-y_ADJUSTMENT; // move up

} // end 10

else if ( 11 == numberOfLibraries)
{
x[0]=xCircle;
y[0]=START_Y;
g.fillOval(x[0],y[0],4,4); // top of circle

labelX[0]=x[0]-x_ADJUSTMENT; // move library name out of circle
labelY[0]=y[0]-y_ADJUSTMENT; // move up

x[1]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
y[1]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
g.fillOval(x[1],y[1],4,4); // 1st quadrant

labelX[1]=x[1]+3;
labelY[1]=y[1]-y_ADJUSTMENT;

x[2]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(27*Math.PI/180));
y[2]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(27*Math.PI/180));

```

```

g.fillOval(x[2],y[2],4,4); // 1st quadrant

labelX[2]=x[2]+3;
labelY[2]=y[2]-y_ADJUSTMENT;

x[3]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[3]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[3],y[3],4,4); // 2nd quadrant

labelX[3]=x[3]+3;
labelY[3]=y[3]+y_ADJUSTMENT;

x[4]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(39*Math.PI/180));
y[4]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(39*Math.PI/180));
g.fillOval(x[4],y[4],4,4); // 2nd quadrant

labelX[4]=x[4]+3;
labelY[4]=y[4]+y_ADJUSTMENT;

x[5]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(72*Math.PI/180));
y[5]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(72*Math.PI/180));
g.fillOval(x[5],y[5],4,4); // 2nd quadrant

labelX[5]=x[5]+3;
labelY[5]=y[5]+y_ADJUSTMENT;

x[6]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(75*Math.PI/180));
y[6]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(75*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 3rd quadrant

labelX[6]=x[6]-x_ADJUSTMENT; // move library name out of circle
labelY[6]=y[6]+y_ADJUSTMENT; // move down

x[7]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[7]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 3rd quadrant

labelX[7]=x[7]-x_ADJUSTMENT; // move library name out of circle
labelY[7]=y[7]+y_ADJUSTMENT; // move down

x[8]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(9*Math.PI/180));
y[8]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(9*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 3rd quadrant

labelX[8]=x[8]-x_ADJUSTMENT; // move library name out of circle
labelY[8]=y[8]+y_ADJUSTMENT; // move down

x[9]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(24*Math.PI/180));
y[9]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(24*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 4th quadrant

labelX[9]=x[9]-x_ADJUSTMENT; // move library name out of circle
labelY[9]=y[9]-y_ADJUSTMENT; // move up

x[10]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(57*Math.PI/180));
y[10]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(57*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 4th quadrant

```

```

    labelX[10]=x[10]-x_ADJUSTMENT;    // move library name out of circle
    labelY[10]=y[10]-y_ADJUSTMENT;    // move up
} // end 11

else if ( 12 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

    x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
    y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);    // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
    y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
    g.fillOval(x[2],y[2],4,4);    // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=START_X+2*CIRCLE_RADIUS_INT;
    y[3]=yCircle;
    g.fillOval(x[3],y[3],4,4);    // right

    labelX[3]=x[3]+3;
    labelY[3]=y[3]+y_ADJUSTMENT;

    x[4]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
    y[4]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
    g.fillOval(x[4],y[4],4,4);    // 2nd quadrant

    labelX[4]=x[4]+3;
    labelY[4]=y[4]+y_ADJUSTMENT;

    x[5]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
    y[5]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
    g.fillOval(x[5],y[5],4,4);    // 2nd quadrant

    labelX[5]=x[5]+3;
    labelY[5]=y[5]+y_ADJUSTMENT;

    x[6]=xCircle;
    y[6]=START_Y+2*CIRCLE_RADIUS_INT;
    g.fillOval(x[6],y[6],4,4);    // bottom of circle

    labelX[6]=x[6]-x_ADJUSTMENT;    // move library name out of circle
    labelY[6]=y[6]+y_ADJUSTMENT;    // move down

```

```

x[7]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
y[7]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 3rd quadrant

labelX[7]=x[7]-x_ADJUSTMENT; // move library name out of circle
labelY[7]=y[7]+y_ADJUSTMENT; // move down

x[8]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[8]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 3rd quadrant

labelX[8]=x[8]-x_ADJUSTMENT; // move library name out of circle
labelY[8]=y[8]+y_ADJUSTMENT; // move down

x[9]=START_X;
y[9]=yCircle;
g.fillOval(x[9],y[9],4,4); // left

labelX[9]=x[9]-x_ADJUSTMENT; // move library name out of circle
labelY[9]=y[9]+y_ADJUSTMENT; // move down

x[10]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[10]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 4th quadrant

labelX[10]=x[10]-x_ADJUSTMENT; // move library name out of circle
labelY[10]=y[10]-y_ADJUSTMENT; // move up

x[11]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
y[11]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 4th quadrant
labelX[11]=x[11]-x_ADJUSTMENT; // move library name out of circle
labelY[11]=y[11]-y_ADJUSTMENT; // move up

} // end 12

else if ( 13 == numberOfLibraries)
{
x[0]=xCircle;
y[0]=START_Y;
g.fillOval(x[0],y[0],4,4); // top of circle

labelX[0]=x[0]-x_ADJUSTMENT; // move library name out of circle
labelY[0]=y[0]-y_ADJUSTMENT; // move up

x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(62*Math.PI/180));
y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(62*Math.PI/180));
g.fillOval(x[1],y[1],4,4); // 1st quadrant

labelX[1]=x[1]+3;
labelY[1]=y[1]-y_ADJUSTMENT;

x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(34*Math.PI/180));
y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(34*Math.PI/180));
g.fillOval(x[2],y[2],4,4); // 1st quadrant

labelX[2]=x[2]+3;

```

```

labelY[2]=y[2]-y_ADJUSTMENT;

x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[3]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[3],y[3],4,4); // 1st quadrant

labelX[3]=x[3]+3;
labelY[3]=y[3]-y_ADJUSTMENT;

x[4]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(22*Math.PI/180));
y[4]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(22*Math.PI/180));
g.fillOval(x[4],y[4],4,4); // 2nd quadrant

labelX[4]=x[4]+3;
labelY[4]=y[4]+y_ADJUSTMENT;

x[5]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(50*Math.PI/180));
y[5]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(50*Math.PI/180));
g.fillOval(x[5],y[5],4,4); // 2nd quadrant

labelX[5]=x[5]+3;
labelY[5]=y[5]+y_ADJUSTMENT;

x[6]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[6]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 2nd quadrant

labelX[6]=x[6]+3;
labelY[6]=y[6]+y_ADJUSTMENT;

x[7]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(74*Math.PI/180));
y[7]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(74*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 3rd quadrant

labelX[7]=x[7]-x_ADJUSTMENT; // move library name out of circle
labelY[7]=y[7]+y_ADJUSTMENT; // move down

x[8]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(46*Math.PI/180));
y[8]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(46*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 3rd quadrant

labelX[8]=x[8]-x_ADJUSTMENT; // move library name out of circle
labelY[8]=y[8]+y_ADJUSTMENT; // move down

x[9]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[9]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 3rd quadrant

labelX[9]=x[9]-x_ADJUSTMENT; // move library name out of circle
labelY[9]=y[9]+y_ADJUSTMENT; // move down

x[10]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(10*Math.PI/180));
y[10]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(10*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 4th quadrant

labelX[10]=x[10]-x_ADJUSTMENT; // move library name out of circle
labelY[10]=y[10]-y_ADJUSTMENT; // move up

```

```

x[11]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(38*Math.PI/180));
y[11]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(38*Math.PI/180));
g.fillOval(x[11],y[11],4,4);      // 4th quadrant

labelX[11]=x[11]-x_ADJUSTMENT;    // move library name out of circle
labelY[11]=y[11]-y_ADJUSTMENT;    // move up

x[12]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[12]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[12],y[12],4,4);      // 4th quadrant

labelX[12]=x[12]-x_ADJUSTMENT;    // move library name out of circle
labelY[12]=y[12]-y_ADJUSTMENT;    // move up

} // end 13

else if ( 14 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

    x[1]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(64*Math.PI/180));
    y[1]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(64*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);    // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(38*Math.PI/180));
    y[2]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(38*Math.PI/180));
    g.fillOval(x[2],y[2],4,4);    // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(12*Math.PI/180));
    y[3]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(12*Math.PI/180));
    g.fillOval(x[3],y[3],4,4);    // 1st quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]-y_ADJUSTMENT;

    x[4]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(14*Math.PI/180));
    y[4]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(14*Math.PI/180));
    g.fillOval(x[4],y[4],4,4);    // 2nd quadrant

    labelX[4]=x[4]+3;
    labelY[4]=y[4]+y_ADJUSTMENT;

    x[5]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(40*Math.PI/180));
    y[5]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(40*Math.PI/180));
    g.fillOval(x[5],y[5],4,4);    // 2nd quadrant

```

```

labelX[5]=x[5]+3;
labelY[5]=y[5]+y_ADJUSTMENT;

x[6]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[6]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 2nd quadrant

labelX[6]=x[6]+3;
labelY[6]=y[6]+y_ADJUSTMENT;

x[7]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(88*Math.PI/180));
y[7]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(88*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 3rd quadrant

labelX[7]=x[7]-x_ADJUSTMENT; // move library name out of circle
labelY[7]=y[7]+y_ADJUSTMENT; // move down

x[8]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(62*Math.PI/180));
y[8]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(62*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 3rd quadrant

labelX[8]=x[8]-x_ADJUSTMENT; // move library name out of circle
labelY[8]=y[8]+y_ADJUSTMENT; // move down

x[9]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(36*Math.PI/180));
y[9]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(36*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 3rd quadrant

labelX[9]=x[9]-x_ADJUSTMENT; // move library name out of circle
labelY[9]=y[9]+y_ADJUSTMENT; // move down

x[10]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(10*Math.PI/180));
y[10]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(10*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 3rd quadrant

labelX[10]=x[10]-x_ADJUSTMENT; // move library name out of circle
labelY[10]=y[10]+y_ADJUSTMENT; // move down

x[11]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(16*Math.PI/180));
y[11]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(16*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 4th quadrant

labelX[11]=x[11]-x_ADJUSTMENT; // move library name out of circle
labelY[11]=y[11]-y_ADJUSTMENT; // move up

x[12]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[12]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[12],y[12],4,4); // 4th quadrant

labelX[12]=x[12]-x_ADJUSTMENT; // move library name out of circle
labelY[12]=y[12]-y_ADJUSTMENT; // move up

x[13]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(68*Math.PI/180));
y[13]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(68*Math.PI/180));
g.fillOval(x[13],y[13],4,4); // 4th quadrant

```

```

    labelX[13]=x[13]-x_ADJUSTMENT;    // move library name out of circle
    labelY[13]=y[13]-y_ADJUSTMENT;    // move up

} // end 14

else if ( 15 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

    x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
    y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);    // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
    y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
    g.fillOval(x[2],y[2],4,4);    // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
    y[3]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
    g.fillOval(x[3],y[3],4,4);    // 1st quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]-y_ADJUSTMENT;

    x[4]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
    y[4]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
    g.fillOval(x[4],y[4],4,4);    // 2nd quadrant

    labelX[4]=x[4]+3;
    labelY[4]=y[4]+y_ADJUSTMENT;

    x[5]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
    y[5]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
    g.fillOval(x[5],y[5],4,4);    // 2nd quadrant

    labelX[5]=x[5]+3;
    labelY[5]=y[5]+y_ADJUSTMENT;

    x[6]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
    y[6]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
    g.fillOval(x[6],y[6],4,4);    // 2nd quadrant

    labelX[6]=x[6]+3;
    labelY[6]=y[6]+y_ADJUSTMENT;

```

```

x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[7]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 2nd quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[8]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 3rd quadrant

labelX[8]=x[8]-x_ADJUSTMENT; // move library name out of circle
labelY[8]=y[8]+y_ADJUSTMENT; // move down

x[9]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[9]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 3rd quadrant

labelX[9]=x[9]-x_ADJUSTMENT; // move library name out of circle
labelY[9]=y[9]+y_ADJUSTMENT; // move down

x[10]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[10]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 3rd quadrant

labelX[10]=x[10]-x_ADJUSTMENT; // move library name out of circle
labelY[10]=y[10]+y_ADJUSTMENT; // move down

x[11]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[11]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 3rd quadrant

labelX[11]=x[11]-x_ADJUSTMENT; // move library name out of circle
labelY[11]=y[11]+y_ADJUSTMENT; // move down

x[12]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[12]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[12],y[12],4,4); // 4th quadrant

labelX[12]=x[12]-x_ADJUSTMENT; // move library name out of circle
labelY[12]=y[12]-y_ADJUSTMENT; // move up

x[13]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[13]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[13],y[13],4,4); // 4th quadrant

labelX[13]=x[13]-x_ADJUSTMENT; // move library name out of circle
labelY[13]=y[13]-y_ADJUSTMENT; // move up

x[14]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[14]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[14],y[14],4,4); // 4th quadrant

labelX[14]=x[14]-x_ADJUSTMENT; // move library name out of circle
labelY[14]=y[14]-y_ADJUSTMENT; // move up

} // end 15

```

```

else if ( 16 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4); // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT; // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT; // move up

    x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(68*Math.PI/180));
    y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(68*Math.PI/180));
    g.fillOval(x[1],y[1],4,4); // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
    y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
    g.fillOval(x[2],y[2],4,4); // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(23*Math.PI/180));
    y[3]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(23*Math.PI/180));
    g.fillOval(x[3],y[3],4,4); // 1st quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]-y_ADJUSTMENT;

    x[4]=START_X+2*CIRCLE_RADIUS_INT;
    y[4]=yCircle;
    g.fillOval(x[4],y[4],4,4); // right

    labelX[4]=x[4]+3;
    labelY[4]=y[4]+y_ADJUSTMENT;

    x[5]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(22*Math.PI/180));
    y[5]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(22*Math.PI/180));
    g.fillOval(x[5],y[5],4,4); // 2nd quadrant

    labelX[5]=x[5]+3;
    labelY[5]=y[5]+y_ADJUSTMENT;

    x[6]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
    y[6]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
    g.fillOval(x[6],y[6],4,4); // 2nd quadrant

    labelX[6]=x[6]+3;
    labelY[6]=y[6]+y_ADJUSTMENT;

    x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(67*Math.PI/180));
    y[7]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(67*Math.PI/180));
    g.fillOval(x[7],y[7],4,4); // 2nd quadrant

    labelX[7]=x[7]+3;

```

```

labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle;
y[8]=START_Y+2*CIRCLE_RADIUS_INT;
g.fillOval(x[8],y[8],4,4); // bottom of circle

labelX[8]=x[8]-x_ADJUSTMENT; // move library name out of circle
labelY[8]=y[8]+y_ADJUSTMENT; // move down

x[9]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(68*Math.PI/180));
y[9]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(68*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 3rd quadrant

labelX[9]=x[9]-x_ADJUSTMENT; // move library name out of circle
labelY[9]=y[9]+y_ADJUSTMENT; // move down

x[10]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
y[10]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 3rd quadrant

labelX[10]=x[10]-x_ADJUSTMENT; // move library name out of circle
labelY[10]=y[10]+y_ADJUSTMENT; // move down

x[11]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(23*Math.PI/180));
y[11]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(23*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 3rd quadrant

labelX[11]=x[11]-x_ADJUSTMENT; // move library name out of circle
labelY[11]=y[11]+y_ADJUSTMENT; // move down

x[12]=START_X;
y[12]=yCircle;
g.fillOval(x[12],y[12],4,4); // left

labelX[12]=x[12]-x_ADJUSTMENT; // move library name out of circle
labelY[12]=y[12]+y_ADJUSTMENT; // move down

x[13]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(22*Math.PI/180));
y[13]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(22*Math.PI/180));
g.fillOval(x[13],y[13],4,4); // 4th quadrant

labelX[13]=x[13]-x_ADJUSTMENT; // move library name out of circle
labelY[13]=y[13]-y_ADJUSTMENT; // move up

x[14]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
y[14]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
g.fillOval(x[14],y[14],4,4); // 4th quadrant

labelX[14]=x[14]-x_ADJUSTMENT; // move library name out of circle
labelY[14]=y[14]-y_ADJUSTMENT; // move up

x[15]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(67*Math.PI/180));
y[15]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(67*Math.PI/180));
g.fillOval(x[15],y[15],4,4); // 4th quadrant

labelX[15]=x[15]-x_ADJUSTMENT; // move library name out of circle
labelY[15]=y[15]-y_ADJUSTMENT; // move up

```

```

} // end 16

else if ( 17 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4); // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT; // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT; // move up

    x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(69*Math.PI/180));
    y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(69*Math.PI/180));
    g.fillOval(x[1],y[1],4,4); // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(48*Math.PI/180));
    y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(48*Math.PI/180));
    g.fillOval(x[2],y[2],4,4); // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(27*Math.PI/180));
    y[3]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(27*Math.PI/180));
    g.fillOval(x[3],y[3],4,4); // 1st quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]-y_ADJUSTMENT;

    x[4]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
    y[4]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
    g.fillOval(x[4],y[4],4,4); // 1st quadrant

    labelX[4]=x[4]+3;
    labelY[4]=y[4]-y_ADJUSTMENT;

    x[5]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(15*Math.PI/180));
    y[5]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(15*Math.PI/180));
    g.fillOval(x[5],y[5],4,4); // 2nd quadrant

    labelX[5]=x[5]+3;
    labelY[5]=y[5]+y_ADJUSTMENT;

    x[6]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(36*Math.PI/180));
    y[6]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(36*Math.PI/180));
    g.fillOval(x[6],y[6],4,4); // 2nd quadrant

    labelX[6]=x[6]+3;
    labelY[6]=y[6]+y_ADJUSTMENT;

    x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(57*Math.PI/180));
    y[7]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(57*Math.PI/180));
    g.fillOval(x[7],y[7],4,4); // 2nd quadrant

```

```

labelX[7]=x[7]+3;
labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[8]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 2nd quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]+y_ADJUSTMENT;

x[9]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(81*Math.PI/180));
y[9]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(81*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 3rd quadrant

labelX[9]=x[9]-x_ADJUSTMENT; // move library name out of circle
labelY[9]=y[9]+y_ADJUSTMENT; // move down

x[10]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
y[10]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 3rd quadrant

labelX[10]=x[10]-x_ADJUSTMENT; // move library name out of circle
labelY[10]=y[10]+y_ADJUSTMENT; // move down

x[11]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(39*Math.PI/180));
y[11]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(39*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 3rd quadrant

labelX[11]=x[11]-x_ADJUSTMENT; // move library name out of circle
labelY[11]=y[11]+y_ADJUSTMENT; // move down

x[12]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[12]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[12],y[12],4,4); // 3rd quadrant

labelX[12]=x[12]-x_ADJUSTMENT; // move library name out of circle
labelY[12]=y[12]+y_ADJUSTMENT; // move down

x[13]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(3*Math.PI/180));
y[13]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(3*Math.PI/180));
g.fillOval(x[13],y[13],4,4); // 4th quadrant

labelX[13]=x[13]-x_ADJUSTMENT; // move library name out of circle
labelY[13]=y[13]-y_ADJUSTMENT; // move up

x[14]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(24*Math.PI/180));
y[14]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(24*Math.PI/180));
g.fillOval(x[14],y[14],4,4); // 4th quadrant

labelX[14]=x[14]-x_ADJUSTMENT; // move library name out of circle
labelY[14]=y[14]-y_ADJUSTMENT; // move up

x[15]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
y[15]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
g.fillOval(x[15],y[15],4,4); // 4th quadrant

```

```

labelX[15]=x[15]-x_ADJUSTMENT;    // move library name out of circle
labelY[15]=y[15]-y_ADJUSTMENT;    // move up

x[16]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[16]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[16],y[16],4,4);      // 4th quadrant

labelX[16]=x[16]-x_ADJUSTMENT;    // move library name out of circle
labelY[16]=y[16]-y_ADJUSTMENT;    // move up

} // end 17

else if ( 18 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

    x[1]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(70*Math.PI/180));
    y[1]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(70*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);    // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(50*Math.PI/180));
    y[2]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(50*Math.PI/180));
    g.fillOval(x[2],y[2],4,4);    // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
    y[3]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
    g.fillOval(x[3],y[3],4,4);    // 1st quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]-y_ADJUSTMENT;

    x[4]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(10*Math.PI/180));
    y[4]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(10*Math.PI/180));
    g.fillOval(x[4],y[4],4,4);    // 1st quadrant

    labelX[4]=x[4]+3;
    labelY[4]=y[4]-y_ADJUSTMENT;

    x[5]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(10*Math.PI/180));
    y[5]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(10*Math.PI/180));
    g.fillOval(x[5],y[5],4,4);    // 2nd quadrant

    labelX[5]=x[5]+3;
    labelY[5]=y[5]+y_ADJUSTMENT;

    x[6]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(30*Math.PI/180));

```

```

y[6]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 2nd quadrant

labelX[6]=x[6]+3;
labelY[6]=y[6]+y_ADJUSTMENT;

x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(50*Math.PI/180));
y[7]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(50*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 2nd quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(70*Math.PI/180));
y[8]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(70*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 2nd quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]+y_ADJUSTMENT;

x[9]=xCircle;
y[9]=START_Y+2*CIRCLE_RADIUS_INT;
g.fillOval(x[9],y[9],4,4); // bottom of circle

labelX[9]=x[9]-x_ADJUSTMENT; // move library name out of circle
labelY[9]=y[9]+y_ADJUSTMENT; // move down

x[10]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(70*Math.PI/180));
y[10]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(70*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 3rd quadrant

labelX[10]=x[10]-x_ADJUSTMENT; // move library name out of circle
labelY[10]=y[10]+y_ADJUSTMENT; // move down

x[11]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(50*Math.PI/180));
y[11]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(50*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 3rd quadrant

labelX[11]=x[11]-x_ADJUSTMENT; // move library name out of circle
labelY[11]=y[11]+y_ADJUSTMENT; // move down

x[12]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[12]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[12],y[12],4,4); // 3rd quadrant

labelX[12]=x[12]-x_ADJUSTMENT; // move library name out of circle
labelY[12]=y[12]+y_ADJUSTMENT; // move down

x[13]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(10*Math.PI/180));
y[13]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(10*Math.PI/180));
g.fillOval(x[13],y[13],4,4); // 3rd quadrant

labelX[13]=x[13]-x_ADJUSTMENT; // move library name out of circle
labelY[13]=y[13]+y_ADJUSTMENT; // move down

x[14]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(10*Math.PI/180));
y[14]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(10*Math.PI/180));

```

```

g.fillOval(x[14],y[14],4,4);      // 4th quadrant

labelX[14]=x[14]-x_ADJUSTMENT;    // move library name out of circle
labelY[14]=y[14]-y_ADJUSTMENT;    // move up

x[15]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[15]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[15],y[15],4,4);      // 4th quadrant

labelX[15]=x[15]-x_ADJUSTMENT;    // move library name out of circle
labelY[15]=y[15]-y_ADJUSTMENT;    // move up

x[16]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(50*Math.PI/180));
y[16]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(50*Math.PI/180));
g.fillOval(x[16],y[16],4,4);      // 4th quadrant

labelX[16]=x[16]-x_ADJUSTMENT;    // move library name out of circle
labelY[16]=y[16]-y_ADJUSTMENT;    // move up

x[17]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(70*Math.PI/180));
y[17]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(70*Math.PI/180));
g.fillOval(x[17],y[17],4,4);      // 4th quadrant

labelX[17]=x[17]-x_ADJUSTMENT;    // move library name out of circle
labelY[17]=y[17]-y_ADJUSTMENT;    // move up

} // end 18

else if ( 19 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;  // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;  // move up

    x[1]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(71*Math.PI/180));
    y[1]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(71*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);    // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(52*Math.PI/180));
    y[2]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(52*Math.PI/180));
    g.fillOval(x[2],y[2],4,4);    // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(33*Math.PI/180));
    y[3]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(33*Math.PI/180));
    g.fillOval(x[3],y[3],4,4);    // 1st quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]-y_ADJUSTMENT;

```

```

x[4]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(14*Math.PI/180));
y[4]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(14*Math.PI/180));
g.fillOval(x[4],y[4],4,4); // 1st quadrant

labelX[4]=x[4]+3;
labelY[4]=y[4]-y_ADJUSTMENT;

x[5]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(5*Math.PI/180));
y[5]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(5*Math.PI/180));
g.fillOval(x[5],y[5],4,4); // 2nd quadrant

labelX[5]=x[5]+3;
labelY[5]=y[5]+y_ADJUSTMENT;

x[6]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(24*Math.PI/180));
y[6]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(24*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 2nd quadrant

labelX[6]=x[6]+3;
labelY[6]=y[6]+y_ADJUSTMENT;

x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(43*Math.PI/180));
y[7]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(43*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 2nd quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(62*Math.PI/180));
y[8]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(62*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 2nd quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]+y_ADJUSTMENT;

x[9]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(81*Math.PI/180));
y[9]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(81*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

x[10]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(80*Math.PI/180));
y[10]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(80*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 3rd quadrant

labelX[10]=x[10]-x_ADJUSTMENT; // move library name out of circle
labelY[10]=y[10]+y_ADJUSTMENT; // move down

x[11]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(61*Math.PI/180));
y[11]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(61*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 3rd quadrant

labelX[11]=x[11]-x_ADJUSTMENT; // move library name out of circle
labelY[11]=y[11]+y_ADJUSTMENT; // move down

```

```

x[12]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[12]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[12],y[12],4,4);          // 3rd quadrant

labelX[12]=x[12]-x_ADJUSTMENT;        // move library name out of circle
labelY[12]=y[12]+y_ADJUSTMENT;        // move down

x[13]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(23*Math.PI/180));
y[13]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(23*Math.PI/180));
g.fillOval(x[13],y[13],4,4);          // 3rd quadrant

labelX[13]=x[13]-x_ADJUSTMENT;        // move library name out of circle
labelY[13]=y[13]+y_ADJUSTMENT;        // move down

x[14]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(4*Math.PI/180));
y[14]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(4*Math.PI/180));
g.fillOval(x[14],y[14],4,4);          // 3rd quadrant

labelX[14]=x[14]-x_ADJUSTMENT;        // move library name out of circle
labelY[14]=y[14]+y_ADJUSTMENT;        // move down

x[15]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(15*Math.PI/180));
y[15]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(15*Math.PI/180));
g.fillOval(x[15],y[15],4,4);          // 4th quadrant

labelX[15]=x[15]-x_ADJUSTMENT;        // move library name out of circle
labelY[15]=y[15]-y_ADJUSTMENT;        // move up

x[16]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(34*Math.PI/180));
y[16]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(34*Math.PI/180));
g.fillOval(x[16],y[16],4,4);          // 4th quadrant

labelX[16]=x[16]-x_ADJUSTMENT;        // move library name out of circle
labelY[16]=y[16]-y_ADJUSTMENT;        // move up

x[17]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(53*Math.PI/180));
y[17]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(53*Math.PI/180));
g.fillOval(x[17],y[17],4,4);          // 4th quadrant

labelX[17]=x[17]-x_ADJUSTMENT;        // move library name out of circle
labelY[17]=y[17]-y_ADJUSTMENT;        // move up

x[18]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(72*Math.PI/180));
y[18]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(72*Math.PI/180));
g.fillOval(x[18],y[18],4,4);          // 4th quadrant

labelX[18]=x[18]-x_ADJUSTMENT;        // move library name out of circle
labelY[18]=y[18]-y_ADJUSTMENT;        // move up

} // end 19

else if ( 20 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);          // top of circle

```

```

labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
labelY[0]=y[0]-y_ADJUSTMENT;    // move up

x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(72*Math.PI/180));
y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(72*Math.PI/180));
g.fillOval(x[1],y[1],4,4);    // 1st quadrant

labelX[1]=x[1]+3;
labelY[1]=y[1]-y_ADJUSTMENT;

x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[2],y[2],4,4);    // 1st quadrant

labelX[2]=x[2]+3;
labelY[2]=y[2]-y_ADJUSTMENT;

x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(36*Math.PI/180));
y[3]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(36*Math.PI/180));
g.fillOval(x[3],y[3],4,4);    // 1st quadrant

labelX[3]=x[3]+3;
labelY[3]=y[3]-y_ADJUSTMENT;

x[4]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[4]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[4],y[4],4,4);    // 1st quadrant

labelX[4]=x[4]+3;
labelY[4]=y[4]-y_ADJUSTMENT;

x[5]=START_X+2*CIRCLE_RADIUS_INT;
y[5]=yCircle;
g.fillOval(x[5],y[5],4,4);    // right

labelX[5]=x[5]+3;
labelY[5]=y[5]+y_ADJUSTMENT;

x[6]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[6]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[6],y[6],4,4);    // 2nd quadrant

labelX[6]=x[6]+3;
labelY[6]=y[6]+y_ADJUSTMENT;

x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(36*Math.PI/180));
y[7]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(36*Math.PI/180));
g.fillOval(x[7],y[7],4,4);    // 2nd quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[8]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[8],y[8],4,4);    // 2nd quadrant

labelX[8]=x[8]+3;

```

```

labelY[8]=y[8]+y_ADJUSTMENT;

x[9]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(72*Math.PI/180));
y[9]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(72*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

x[10]=xCircle;
y[10]=START_Y+2*CIRCLE_RADIUS_INT;
g.fillOval(x[10],y[10],4,4); // bottom of circle

labelX[10]=x[10]-x_ADJUSTMENT; // move library name out of circle
labelY[10]=y[10]+y_ADJUSTMENT; // move down

x[11]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(72*Math.PI/180));
y[11]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(72*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 3rd quadrant

labelX[11]=x[11]-x_ADJUSTMENT; // move library name out of circle
labelY[11]=y[11]+y_ADJUSTMENT; // move down

x[12]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[12]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[12],y[12],4,4); // 3rd quadrant

labelX[12]=x[12]-x_ADJUSTMENT; // move library name out of circle
labelY[12]=y[12]+y_ADJUSTMENT; // move down

x[13]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(36*Math.PI/180));
y[13]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(36*Math.PI/180));
g.fillOval(x[13],y[13],4,4); // 3rd quadrant

labelX[13]=x[13]-x_ADJUSTMENT; // move library name out of circle
labelY[13]=y[13]+y_ADJUSTMENT; // move down

x[14]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[14]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[14],y[14],4,4); // 3rd quadrant

labelX[14]=x[14]-x_ADJUSTMENT; // move library name out of circle
labelY[14]=y[14]+y_ADJUSTMENT; // move down

x[15]=START_X;
y[15]=yCircle;
g.fillOval(x[15],y[15],4,4); // left

labelX[15]=x[15]-x_ADJUSTMENT; // move library name out of circle
labelY[15]=y[15]+y_ADJUSTMENT; // move down

x[16]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[16]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[16],y[16],4,4); // 4th quadrant

labelX[16]=x[16]-x_ADJUSTMENT; // move library name out of circle
labelY[16]=y[16]-y_ADJUSTMENT; // move up

```

```

x[17]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(36*Math.PI/180));
y[17]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(36*Math.PI/180));
g.fillOval(x[17],y[17],4,4); // 4th quadrant

labelX[17]=x[17]-x_ADJUSTMENT; // move library name out of circle
labelY[17]=y[17]-y_ADJUSTMENT; // move up

x[18]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[18]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[18],y[18],4,4); // 4th quadrant

labelX[18]=x[18]-x_ADJUSTMENT; // move library name out of circle
labelY[18]=y[18]-y_ADJUSTMENT; // move up

x[19]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(72*Math.PI/180));
y[19]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(72*Math.PI/180));
g.fillOval(x[19],y[19],4,4); // 4th quadrant

labelX[19]=x[19]-x_ADJUSTMENT; // move library name out of circle
labelY[19]=y[19]-y_ADJUSTMENT; // move up

} // end 20

else if ( 21 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4); // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT; // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT; // move up

    x[1]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(73*Math.PI/180));
    y[1]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(73*Math.PI/180));
    g.fillOval(x[1],y[1],4,4); // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(56*Math.PI/180));
    y[2]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(56*Math.PI/180));
    g.fillOval(x[2],y[2],4,4); // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(39*Math.PI/180));
    y[3]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(39*Math.PI/180));
    g.fillOval(x[3],y[3],4,4); // 1st quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]-y_ADJUSTMENT;

    x[4]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(22*Math.PI/180));
    y[4]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(22*Math.PI/180));
    g.fillOval(x[4],y[4],4,4); // 1st quadrant

```

```

labelX[4]=x[4]+3;
labelY[4]=y[4]-y_ADJUSTMENT;

x[5]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(5*Math.PI/180));
y[5]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(5*Math.PI/180));
g.fillOval(x[5],y[5],4,4); // 1st quadrant

labelX[5]=x[5]+3;
labelY[5]=y[5]-y_ADJUSTMENT;

x[6]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(12*Math.PI/180));
y[6]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(12*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 2nd quadrant

labelX[6]=x[6]+3;
labelY[6]=y[6]+y_ADJUSTMENT;

x[7]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(29*Math.PI/180));
y[7]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(29*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 2nd quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(46*Math.PI/180));
y[8]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(46*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 2nd quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]+y_ADJUSTMENT;

x[9]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(63*Math.PI/180));
y[9]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(63*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

x[10]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(80*Math.PI/180));
y[10]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(80*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 2nd quadrant

labelX[10]=x[10]+3;
labelY[10]=y[10]+y_ADJUSTMENT;

x[11]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(83*Math.PI/180));
y[11]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(83*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 3rd quadrant

labelX[11]=x[11]-x_ADJUSTMENT; // move library name out of circle
labelY[11]=y[11]+y_ADJUSTMENT; // move down

x[12]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[12]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[12],y[12],4,4); // 3rd quadrant

```

```

labelX[12]=x[12]-x_ADJUSTMENT;    // move library name out of circle
labelY[12]=y[12]+y_ADJUSTMENT;    // move down

x[13]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(49*Math.PI/180));
y[13]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(49*Math.PI/180));
g.fillOval(x[13],y[13],4,4);      // 3rd quadrant

labelX[13]=x[13]-x_ADJUSTMENT;    // move library name out of circle
labelY[13]=y[13]+y_ADJUSTMENT;    // move down

x[14]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(32*Math.PI/180));
y[14]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(32*Math.PI/180));
g.fillOval(x[14],y[14],4,4);      // 3rd quadrant

labelX[14]=x[14]-x_ADJUSTMENT;    // move library name out of circle
labelY[14]=y[14]+y_ADJUSTMENT;    // move down

x[15]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(15*Math.PI/180));
y[15]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(15*Math.PI/180));
g.fillOval(x[15],y[15],4,4);      // 3rd quadrant

labelX[15]=x[15]-x_ADJUSTMENT;    // move library name out of circle
labelY[15]=y[15]+y_ADJUSTMENT;    // move down

x[16]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(2*Math.PI/180));
y[16]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(2*Math.PI/180));
g.fillOval(x[16],y[16],4,4);      // 4th quadrant

labelX[16]=x[16]-x_ADJUSTMENT;    // move library name out of circle
labelY[16]=y[16]-y_ADJUSTMENT;    // move up

x[17]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(19*Math.PI/180));
y[17]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(19*Math.PI/180));
g.fillOval(x[17],y[17],4,4);      // 4th quadrant

labelX[17]=x[17]-x_ADJUSTMENT;    // move library name out of circle
labelY[17]=y[17]-y_ADJUSTMENT;    // move up

x[18]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(36*Math.PI/180));
y[18]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(36*Math.PI/180));
g.fillOval(x[18],y[18],4,4);      // 4th quadrant

labelX[18]=x[18]-x_ADJUSTMENT;    // move library name out of circle
labelY[18]=y[18]-y_ADJUSTMENT;    // move up

x[19]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(53*Math.PI/180));
y[19]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(53*Math.PI/180));
g.fillOval(x[19],y[19],4,4);      // 4th quadrant

labelX[19]=x[19]-x_ADJUSTMENT;    // move library name out of circle
labelY[19]=y[19]-y_ADJUSTMENT;    // move up

x[20]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(70*Math.PI/180));
y[20]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(70*Math.PI/180));
g.fillOval(x[20],y[20],4,4);      // 4th quadrant

labelX[20]=x[20]-x_ADJUSTMENT;    // move library name out of circle

```

```

        labelY[20]=y[20]-y_ADJUSTMENT;    // move up
    } // end 21

else if ( 22 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;        // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

    x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(74*Math.PI/180));
    y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(74*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);    // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(58*Math.PI/180));
    y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(58*Math.PI/180));
    g.fillOval(x[2],y[2],4,4);    // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
    y[3]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
    g.fillOval(x[3],y[3],4,4);    // 1st quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]-y_ADJUSTMENT;

    x[4]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(26*Math.PI/180));
    y[4]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(26*Math.PI/180));
    g.fillOval(x[4],y[4],4,4);    // 1st quadrant

    labelX[4]=x[4]+3;
    labelY[4]=y[4]-y_ADJUSTMENT;

    x[5]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(10*Math.PI/180));
    y[5]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(10*Math.PI/180));
    g.fillOval(x[5],y[5],4,4);    // 1st quadrant

    labelX[5]=x[5]+3;
    labelY[5]=y[5]-y_ADJUSTMENT;

    x[6]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
    y[6]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
    g.fillOval(x[6],y[6],4,4);    // 2nd quadrant

    labelX[6]=x[6]+3;
    labelY[6]=y[6]+y_ADJUSTMENT;

    x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(22*Math.PI/180));
    y[7]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(22*Math.PI/180));

```

```

g.fillOval(x[7],y[7],4,4); // 2nd quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(38*Math.PI/180));
y[8]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(38*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 2nd quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]+y_ADJUSTMENT;

x[9]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[9]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

x[10]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(70*Math.PI/180));
y[10]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(70*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 2nd quadrant

labelX[10]=x[10]+3;
labelY[10]=y[10]+y_ADJUSTMENT;

x[11]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(86*Math.PI/180));
y[11]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(86*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 2nd quadrant

labelX[11]=x[11]+3;
labelY[11]=y[11]+y_ADJUSTMENT;

x[12]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[12]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[12],y[12],4,4); // 3rd quadrant

labelX[12]=x[12]-x_ADJUSTMENT; // move library name out of circle
labelY[12]=y[12]+y_ADJUSTMENT; // move down

x[13]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(62*Math.PI/180));
y[13]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(62*Math.PI/180));
g.fillOval(x[13],y[13],4,4); // 3rd quadrant

labelX[13]=x[13]-x_ADJUSTMENT; // move library name out of circle
labelY[13]=y[13]+y_ADJUSTMENT; // move down

x[14]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(46*Math.PI/180));
y[14]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(46*Math.PI/180));
g.fillOval(x[14],y[14],4,4); // 3rd quadrant

labelX[14]=x[14]-x_ADJUSTMENT; // move library name out of circle
labelY[14]=y[14]+y_ADJUSTMENT; // move down

x[15]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(29*Math.PI/180));
y[15]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(29*Math.PI/180));
g.fillOval(x[15],y[15],4,4); // 3rd quadrant

```

```

labelX[15]=x[15]-x_ADJUSTMENT;    // move library name out of circle
labelY[15]=y[15]+y_ADJUSTMENT;    // move down

x[16]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(12*Math.PI/180));
y[16]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(12*Math.PI/180));
g.fillOval(x[16],y[16],4,4);      // 3rd quadrant

labelX[16]=x[16]-x_ADJUSTMENT;    // move library name out of circle
labelY[16]=y[16]+y_ADJUSTMENT;    // move down

x[17]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(5*Math.PI/180));
y[17]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(5*Math.PI/180));
g.fillOval(x[17],y[17],4,4);      // 4th quadrant

labelX[17]=x[17]-x_ADJUSTMENT;    // move library name out of circle
labelY[17]=y[17]-y_ADJUSTMENT;    // move up

x[18]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(22*Math.PI/180));
y[18]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(22*Math.PI/180));
g.fillOval(x[18],y[18],4,4);      // 4th quadrant

labelX[18]=x[18]-x_ADJUSTMENT;    // move library name out of circle
labelY[18]=y[18]-y_ADJUSTMENT;    // move up

x[19]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(39*Math.PI/180));
y[19]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(39*Math.PI/180));
g.fillOval(x[19],y[19],4,4);      // 4th quadrant

labelX[19]=x[19]-x_ADJUSTMENT;    // move library name out of circle
labelY[19]=y[19]-y_ADJUSTMENT;    // move up

x[20]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(56*Math.PI/180));
y[20]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(56*Math.PI/180));
g.fillOval(x[20],y[20],4,4);      // 4th quadrant

labelX[20]=x[20]-x_ADJUSTMENT;    // move library name out of circle
labelY[20]=y[20]-y_ADJUSTMENT;    // move up

x[21]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(73*Math.PI/180));
y[21]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(73*Math.PI/180));
g.fillOval(x[21],y[21],4,4);      // 4th quadrant

labelX[21]=x[21]-x_ADJUSTMENT;    // move library name out of circle
labelY[21]=y[21]-y_ADJUSTMENT;    // move up

} // end 22

else if ( 23 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

```

```

x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(74*Math.PI/180));
y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(74*Math.PI/180));
g.fillOval(x[1],y[1],4,4); // 1st quadrant

labelX[1]=x[1]+3;
labelY[1]=y[1]-y_ADJUSTMENT;

x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(58*Math.PI/180));
y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(58*Math.PI/180));
g.fillOval(x[2],y[2],4,4); // 1st quadrant

labelX[2]=x[2]+3;
labelY[2]=y[2]-y_ADJUSTMENT;

x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[3]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[3],y[3],4,4); // 1st quadrant

labelX[3]=x[3]+3;
labelY[3]=y[3]-y_ADJUSTMENT;

x[4]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(26*Math.PI/180));
y[4]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(26*Math.PI/180));
g.fillOval(x[4],y[4],4,4); // 1st quadrant

labelX[4]=x[4]+3;
labelY[4]=y[4]-y_ADJUSTMENT;

x[5]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(10*Math.PI/180));
y[5]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(10*Math.PI/180));
g.fillOval(x[5],y[5],4,4); // 1st quadrant

labelX[5]=x[5]+3;
labelY[5]=y[5]-y_ADJUSTMENT;

x[6]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[6]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 2nd quadrant

labelX[6]=x[6]+3;
labelY[6]=y[6]+y_ADJUSTMENT;

x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(22*Math.PI/180));
y[7]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(22*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 2nd quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(38*Math.PI/180));
y[8]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(38*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 2nd quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]+y_ADJUSTMENT;

x[9]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));

```

```

y[9]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

x[10]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(70*Math.PI/180));
y[10]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(70*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 2nd quadrant

labelX[10]=x[10]+3;
labelY[10]=y[10]+y_ADJUSTMENT;

x[11]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(86*Math.PI/180));
y[11]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(86*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 2nd quadrant

labelX[11]=x[11]+3;
labelY[11]=y[11]+y_ADJUSTMENT;

x[12]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[12]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[12],y[12],4,4); // 3rd quadrant

labelX[12]=x[12]-x_ADJUSTMENT; // move library name out of circle
labelY[12]=y[12]+y_ADJUSTMENT; // move down

x[13]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(62*Math.PI/180));
y[13]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(62*Math.PI/180));
g.fillOval(x[13],y[13],4,4); // 3rd quadrant

labelX[13]=x[13]-x_ADJUSTMENT; // move library name out of circle
labelY[13]=y[13]+y_ADJUSTMENT; // move down

x[14]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(46*Math.PI/180));
y[14]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(46*Math.PI/180));
g.fillOval(x[14],y[14],4,4); // 3rd quadrant

labelX[14]=x[14]-x_ADJUSTMENT; // move library name out of circle
labelY[14]=y[14]+y_ADJUSTMENT; // move down

x[15]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[15]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[15],y[15],4,4); // 3rd quadrant

labelX[15]=x[15]-x_ADJUSTMENT; // move library name out of circle
labelY[15]=y[15]+y_ADJUSTMENT; // move down

x[16]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(15*Math.PI/180));
y[16]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(15*Math.PI/180));
g.fillOval(x[16],y[16],4,4); // 3rd quadrant

labelX[16]=x[16]-x_ADJUSTMENT; // move library name out of circle
labelY[16]=y[16]+y_ADJUSTMENT; // move down

x[17]=START_X;
y[17]=yCircle;

```

```

g.fillOval(x[17],y[17],4,4);      // left

labelX[17]=x[17]-x_ADJUSTMENT;    // move library name out of circle
labelY[17]=y[17]+y_ADJUSTMENT;    // move down

x[18]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(15*Math.PI/180));
y[18]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(15*Math.PI/180));
g.fillOval(x[18],y[18],4,4);      // 4th quadrant

labelX[18]=x[18]-x_ADJUSTMENT;    // move library name out of circle
labelY[18]=y[18]-y_ADJUSTMENT;    // move up

x[19]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[19]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[19],y[19],4,4);      // 4th quadrant

labelX[19]=x[19]-x_ADJUSTMENT;    // move library name out of circle
labelY[19]=y[19]-y_ADJUSTMENT;    // move up

x[20]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
y[20]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
g.fillOval(x[20],y[20],4,4);      // 4th quadrant

labelX[20]=x[20]-x_ADJUSTMENT;    // move library name out of circle
labelY[20]=y[20]-y_ADJUSTMENT;    // move up

x[21]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
y[21]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
g.fillOval(x[21],y[21],4,4);      // 4th quadrant

labelX[21]=x[21]-x_ADJUSTMENT;    // move library name out of circle
labelY[21]=y[21]-y_ADJUSTMENT;    // move up

x[22]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(75*Math.PI/180));
y[22]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(75*Math.PI/180));
g.fillOval(x[22],y[22],4,4);      // 4th quadrant

labelX[22]=x[22]-x_ADJUSTMENT;    // move library name out of circle
labelY[22]=y[22]-y_ADJUSTMENT;    // move up

} // end 23

else if ( 24 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);      // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

    x[1]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(75*Math.PI/180));
    y[1]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(75*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);      // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

```

```

x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
g.fillOval(x[2],y[2],4,4); // 1st quadrant

labelX[2]=x[2]+3;
labelY[2]=y[2]-y_ADJUSTMENT;

x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
y[3]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
g.fillOval(x[3],y[3],4,4); // 1st quadrant

labelX[3]=x[3]+3;
labelY[3]=y[3]-y_ADJUSTMENT;

x[4]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[4]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[4],y[4],4,4); // 1st quadrant

labelX[4]=x[4]+3;
labelY[4]=y[4]-y_ADJUSTMENT;

x[5]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(15*Math.PI/180));
y[5]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(15*Math.PI/180));
g.fillOval(x[5],y[5],4,4); // 1st quadrant

labelX[5]=x[5]+3;
labelY[5]=y[5]-y_ADJUSTMENT;

x[6]=START_X+2*CIRCLE_RADIUS_INT;
y[6]=yCircle;
g.fillOval(x[6],y[6],4,4); // right

labelX[6]=x[6]+3;
labelY[6]=y[6]+y_ADJUSTMENT;

x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(15*Math.PI/180));
y[7]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(15*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 2nd quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[8]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 2nd quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]+y_ADJUSTMENT;

x[9]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
y[9]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

```

```

x[10]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
y[10]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
g.fillOval(x[10],y[10],4,4);          // 2nd quadrant

labelX[10]=x[10]+3;
labelY[10]=y[10]+y_ADJUSTMENT;

x[11]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(75*Math.PI/180));
y[11]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(75*Math.PI/180));
g.fillOval(x[11],y[11],4,4);          // 2nd quadrant

labelX[11]=x[11]+3;
labelY[11]=y[11]+y_ADJUSTMENT;

x[12]=xCircle;
y[12]=START_Y+2*CIRCLE_RADIUS_INT;
g.fillOval(x[12],y[12],4,4);          // bottom of circle

labelX[12]=x[12]-x_ADJUSTMENT;        // move library name out of circle
labelY[12]=y[12]+y_ADJUSTMENT;        // move down

x[13]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(75*Math.PI/180));
y[13]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(75*Math.PI/180));
g.fillOval(x[13],y[13],4,4);          // 3rd quadrant

labelX[13]=x[13]-x_ADJUSTMENT;        // move library name out of circle
labelY[13]=y[13]+y_ADJUSTMENT;        // move down

x[14]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
y[14]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
g.fillOval(x[14],y[14],4,4);          // 3rd quadrant

labelX[14]=x[14]-x_ADJUSTMENT;        // move library name out of circle
labelY[14]=y[14]+y_ADJUSTMENT;        // move down

x[15]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
y[15]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
g.fillOval(x[15],y[15],4,4);          // 3rd quadrant

labelX[15]=x[15]-x_ADJUSTMENT;        // move library name out of circle
labelY[15]=y[15]+y_ADJUSTMENT;        // move down

x[16]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[16]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[16],y[16],4,4);          // 3rd quadrant

labelX[16]=x[16]-x_ADJUSTMENT;        // move library name out of circle
labelY[16]=y[16]+y_ADJUSTMENT;        // move down

x[17]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(15*Math.PI/180));
y[17]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(15*Math.PI/180));
g.fillOval(x[17],y[17],4,4);          // 3rd quadrant

labelX[17]=x[17]-x_ADJUSTMENT;        // move library name out of circle
labelY[17]=y[17]+y_ADJUSTMENT;        // move down

x[18]=START_X;

```

```

y[18]=yCircle;
g.fillOval(x[18],y[18],4,4);      // left

labelX[18]=x[18]-x_ADJUSTMENT;    // move library name out of circle
labelY[18]=y[18]+y_ADJUSTMENT;    // move down

x[19]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(15*Math.PI/180));
y[19]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(15*Math.PI/180));
g.fillOval(x[19],y[19],4,4);      // 4th quadrant

labelX[19]=x[19]-x_ADJUSTMENT;    // move library name out of circle
labelY[19]=y[19]-y_ADJUSTMENT;    // move up

x[20]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[20]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[20],y[20],4,4);      // 4th quadrant

labelX[20]=x[20]-x_ADJUSTMENT;    // move library name out of circle
labelY[20]=y[20]-y_ADJUSTMENT;    // move up

x[21]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
y[21]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
g.fillOval(x[21],y[21],4,4);      // 4th quadrant

labelX[21]=x[21]-x_ADJUSTMENT;    // move library name out of circle
labelY[21]=y[21]-y_ADJUSTMENT;    // move up

x[22]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
y[22]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
g.fillOval(x[22],y[22],4,4);      // 4th quadrant

labelX[22]=x[22]-x_ADJUSTMENT;    // move library name out of circle
labelY[22]=y[22]-y_ADJUSTMENT;    // move up

x[23]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(75*Math.PI/180));
y[23]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(75*Math.PI/180));
g.fillOval(x[23],y[23],4,4);      // 4th quadrant

labelX[23]=x[23]-x_ADJUSTMENT;    // move library name out of circle
labelY[23]=y[23]-y_ADJUSTMENT;    // move up

} // end 24

else if ( 25 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;  // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;  // move up

    x[1]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(75*Math.PI/180));
    y[1]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(75*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);    // 1st quadrant

    labelX[1]=x[1]+3;

```

```

labelY[1]=y[1]-y_ADJUSTMENT;

x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
g.fillOval(x[2],y[2],4,4); // 1st quadrant

labelX[2]=x[2]+3;
labelY[2]=y[2]-y_ADJUSTMENT;

x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
y[3]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
g.fillOval(x[3],y[3],4,4); // 1st quadrant

labelX[3]=x[3]+3;
labelY[3]=y[3]-y_ADJUSTMENT;

x[4]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[4]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[4],y[4],4,4); // 1st quadrant

labelX[4]=x[4]+3;
labelY[4]=y[4]-y_ADJUSTMENT;

x[5]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(15*Math.PI/180));
y[5]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(15*Math.PI/180));
g.fillOval(x[5],y[5],4,4); // 1st quadrant

labelX[5]=x[5]+3;
labelY[5]=y[5]-y_ADJUSTMENT;

x[6]=START_X+2*CIRCLE_RADIUS_INT;
y[6]=yCircle;
g.fillOval(x[6],y[6],4,4); // right

labelX[6]=x[6]+3;
labelY[6]=y[6]+y_ADJUSTMENT;

x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(15*Math.PI/180));
y[7]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(15*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 2nd quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[8]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 2nd quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]+y_ADJUSTMENT;

x[9]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(45*Math.PI/180));
y[9]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(45*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

```

```

x[10]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
y[10]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
g.fillOval(x[10],y[10],4,4);          // 2nd quadrant

labelX[10]=x[10]+3;
labelY[10]=y[10]+y_ADJUSTMENT;

x[11]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(74*Math.PI/180));
y[11]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(74*Math.PI/180));
g.fillOval(x[11],y[11],4,4);          // 2nd quadrant

labelX[11]=x[11]+3;
labelY[11]=y[11]+y_ADJUSTMENT;

x[12]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(88*Math.PI/180));
y[12]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(88*Math.PI/180));
g.fillOval(x[12],y[12],4,4);          // 2nd quadrant

labelX[12]=x[12]+3;
labelY[12]=y[12]+y_ADJUSTMENT;

x[13]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[13]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[13],y[13],4,4);          // 3rd quadrant

labelX[13]=x[13]-x_ADJUSTMENT;         // move library name out of circle
labelY[13]=y[13]+y_ADJUSTMENT;         // move down

x[14]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(64*Math.PI/180));
y[14]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(64*Math.PI/180));
g.fillOval(x[14],y[14],4,4);          // 3rd quadrant

labelX[14]=x[14]-x_ADJUSTMENT;         // move library name out of circle
labelY[14]=y[14]+y_ADJUSTMENT;         // move down

x[15]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(50*Math.PI/180));
y[15]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(50*Math.PI/180));
g.fillOval(x[15],y[15],4,4);          // 3rd quadrant

labelX[15]=x[15]-x_ADJUSTMENT;         // move library name out of circle
labelY[15]=y[15]+y_ADJUSTMENT;         // move down

x[16]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(36*Math.PI/180));
y[16]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(36*Math.PI/180));
g.fillOval(x[16],y[16],4,4);          // 3rd quadrant

labelX[16]=x[16]-x_ADJUSTMENT;         // move library name out of circle
labelY[16]=y[16]+y_ADJUSTMENT;         // move down

x[17]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(22*Math.PI/180));
y[17]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(22*Math.PI/180));
g.fillOval(x[17],y[17],4,4);          // 3rd quadrant

labelX[17]=x[17]-x_ADJUSTMENT;         // move library name out of circle
labelY[17]=y[17]+y_ADJUSTMENT;         // move down

```

```

x[18]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(8*Math.PI/180));
y[18]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(8*Math.PI/180));
g.fillOval(x[18],y[18],4,4);      // 3rd quadrant

labelX[18]=x[18]-x_ADJUSTMENT;    // move library name out of circle
labelY[18]=y[18]+y_ADJUSTMENT;    // move down

x[19]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[19]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[19],y[19],4,4);      // 4th quadrant

labelX[19]=x[19]-x_ADJUSTMENT;    // move library name out of circle
labelY[19]=y[19]-y_ADJUSTMENT;    // move up

x[20]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(20*Math.PI/180));
y[20]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(20*Math.PI/180));
g.fillOval(x[20],y[20],4,4);      // 4th quadrant

labelX[20]=x[20]-x_ADJUSTMENT;    // move library name out of circle
labelY[20]=y[20]-y_ADJUSTMENT;    // move up

x[21]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(34*Math.PI/180));
y[21]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(34*Math.PI/180));
g.fillOval(x[21],y[21],4,4);      // 4th quadrant

labelX[21]=x[21]-x_ADJUSTMENT;    // move library name out of circle
labelY[21]=y[21]-y_ADJUSTMENT;    // move up

x[22]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(48*Math.PI/180));
y[22]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(48*Math.PI/180));
g.fillOval(x[22],y[22],4,4);      // 4th quadrant

labelX[22]=x[22]-x_ADJUSTMENT;    // move library name out of circle
labelY[22]=y[22]-y_ADJUSTMENT;    // move up

x[23]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(62*Math.PI/180));
y[23]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(62*Math.PI/180));
g.fillOval(x[23],y[23],4,4);      // 4th quadrant

labelX[23]=x[23]-x_ADJUSTMENT;    // move library name out of circle
labelY[23]=y[23]-y_ADJUSTMENT;    // move up

x[24]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(76*Math.PI/180));
y[24]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(76*Math.PI/180));
g.fillOval(x[24],y[24],4,4);      // 4th quadrant

labelX[24]=x[24]-x_ADJUSTMENT;    // move library name out of circle
labelY[24]=y[24]-y_ADJUSTMENT;    // move up

} // end 25

else if ( 26 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

```

```

labelX[0]=x[0]-x_ADJUSTMENT;      // move library name out of circle
labelY[0]=y[0]-y_ADJUSTMENT;    // move up

x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(76*Math.PI/180));
y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(76*Math.PI/180));
g.fillOval(x[1],y[1],4,4);  // 1st quadrant

labelX[1]=x[1]+3;
labelY[1]=y[1]-y_ADJUSTMENT;

x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(62*Math.PI/180));
y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(62*Math.PI/180));
g.fillOval(x[2],y[2],4,4);  // 1st quadrant

labelX[2]=x[2]+3;
labelY[2]=y[2]-y_ADJUSTMENT;

x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(48*Math.PI/180));
y[3]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(48*Math.PI/180));
g.fillOval(x[3],y[3],4,4);  // 1st quadrant

labelX[3]=x[3]+3;
labelY[3]=y[3]-y_ADJUSTMENT;

x[4]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(34*Math.PI/180));
y[4]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(34*Math.PI/180));
g.fillOval(x[4],y[4],4,4);  // 1st quadrant

labelX[4]=x[4]+3;
labelY[4]=y[4]-y_ADJUSTMENT;

x[5]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(20*Math.PI/180));
y[5]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(20*Math.PI/180));
g.fillOval(x[5],y[5],4,4);  // 1st quadrant

labelX[5]=x[5]+3;
labelY[5]=y[5]-y_ADJUSTMENT;

x[6]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[6]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[6],y[6],4,4);  // 1st quadrant

labelX[6]=x[6]+3;
labelY[6]=y[6]-y_ADJUSTMENT;

x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(8*Math.PI/180));
y[7]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(8*Math.PI/180));
g.fillOval(x[7],y[7],4,4);  // 2nd quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(22*Math.PI/180));
y[8]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(22*Math.PI/180));
g.fillOval(x[8],y[8],4,4);  // 2nd quadrant

labelX[8]=x[8]+3;

```

```

labelY[8]=y[8]+y_ADJUSTMENT;

x[9]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(36*Math.PI/180));
y[9]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(36*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

x[10]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(50*Math.PI/180));
y[10]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(50*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 2nd quadrant

labelX[10]=x[10]+3;
labelY[10]=y[10]+y_ADJUSTMENT;

x[11]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(64*Math.PI/180));
y[11]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(64*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 2nd quadrant

labelX[11]=x[11]+3;
labelY[11]=y[11]+y_ADJUSTMENT;

x[12]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[12]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[12],y[12],4,4); // 2nd quadrant

labelX[12]=x[12]+3;
labelY[12]=y[12]+y_ADJUSTMENT;

x[13]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(88*Math.PI/180));
y[13]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(88*Math.PI/180));
g.fillOval(x[13],y[13],4,4); // 3rd quadrant

labelX[13]=x[13]-x_ADJUSTMENT; // move library name out of circle
labelY[13]=y[13]+y_ADJUSTMENT; // move down

x[14]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(74*Math.PI/180));
y[14]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(74*Math.PI/180));
g.fillOval(x[14],y[14],4,4); // 3rd quadrant

labelX[14]=x[14]-x_ADJUSTMENT; // move library name out of circle
labelY[14]=y[14]+y_ADJUSTMENT; // move down

x[15]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
y[15]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
g.fillOval(x[15],y[15],4,4); // 3rd quadrant

labelX[15]=x[15]-x_ADJUSTMENT; // move library name out of circle
labelY[15]=y[15]+y_ADJUSTMENT; // move down

x[16]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(46*Math.PI/180));
y[16]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(46*Math.PI/180));
g.fillOval(x[16],y[16],4,4); // 3rd quadrant

labelX[16]=x[16]-x_ADJUSTMENT; // move library name out of circle
labelY[16]=y[16]+y_ADJUSTMENT; // move down

```

```

x[17]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(32*Math.PI/180));
y[17]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(32*Math.PI/180));
g.fillOval(x[17],y[17],4,4);      // 3rd quadrant

labelX[17]=x[17]-x_ADJUSTMENT;    // move library name out of circle
labelY[17]=y[17]+y_ADJUSTMENT;    // move down

x[18]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[18]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[18],y[18],4,4);      // 3rd quadrant

labelX[18]=x[18]-x_ADJUSTMENT;    // move library name out of circle
labelY[18]=y[18]+y_ADJUSTMENT;    // move down

x[19]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(4*Math.PI/180));
y[19]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(4*Math.PI/180));
g.fillOval(x[19],y[19],4,4);      // 3rd quadrant

labelX[19]=x[19]-x_ADJUSTMENT;    // move library name out of circle
labelY[19]=y[19]+y_ADJUSTMENT;    // move down

x[20]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(10*Math.PI/180));
y[20]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(10*Math.PI/180));
g.fillOval(x[20],y[20],4,4);      // 4th quadrant

labelX[20]=x[20]-x_ADJUSTMENT;    // move library name out of circle
labelY[20]=y[20]-y_ADJUSTMENT;    // move up

x[21]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(24*Math.PI/180));
y[21]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(24*Math.PI/180));
g.fillOval(x[21],y[21],4,4);      // 4th quadrant

labelX[21]=x[21]-x_ADJUSTMENT;    // move library name out of circle
labelY[21]=y[21]-y_ADJUSTMENT;    // move up

x[22]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(38*Math.PI/180));
y[22]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(38*Math.PI/180));
g.fillOval(x[22],y[22],4,4);      // 4th quadrant

labelX[22]=x[22]-x_ADJUSTMENT;    // move library name out of circle
labelY[22]=y[22]-y_ADJUSTMENT;    // move up

x[23]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(51*Math.PI/180));
y[23]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(51*Math.PI/180));
g.fillOval(x[23],y[23],4,4);      // 4th quadrant

labelX[23]=x[23]-x_ADJUSTMENT;    // move library name out of circle
labelY[23]=y[23]-y_ADJUSTMENT;    // move up

x[24]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(64*Math.PI/180));
y[24]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(64*Math.PI/180));
g.fillOval(x[24],y[24],4,4);      // 4th quadrant

labelX[24]=x[24]-x_ADJUSTMENT;    // move library name out of circle
labelY[24]=y[24]-y_ADJUSTMENT;    // move up

```

```

x[25]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(77*Math.PI/180));
y[25]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(77*Math.PI/180));
g.fillOval(x[25],y[25],4,4);      // 4th quadrant

labelX[25]=x[25]-x_ADJUSTMENT;    // move library name out of circle
labelY[25]=y[25]-y_ADJUSTMENT;    // move up

} // end 26

else if ( 27 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);  // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

    x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(76*Math.PI/180));
    y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(76*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);  // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(62*Math.PI/180));
    y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(62*Math.PI/180));
    g.fillOval(x[2],y[2],4,4);  // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(48*Math.PI/180));
    y[3]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(48*Math.PI/180));
    g.fillOval(x[3],y[3],4,4);  // 1st quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]-y_ADJUSTMENT;

    x[4]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(34*Math.PI/180));
    y[4]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(34*Math.PI/180));
    g.fillOval(x[4],y[4],4,4);  // 1st quadrant

    labelX[4]=x[4]+3;
    labelY[4]=y[4]-y_ADJUSTMENT;

    x[5]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(20*Math.PI/180));
    y[5]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(20*Math.PI/180));
    g.fillOval(x[5],y[5],4,4);  // 1st quadrant

    labelX[5]=x[5]+3;
    labelY[5]=y[5]-y_ADJUSTMENT;

    x[6]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
    y[6]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
    g.fillOval(x[6],y[6],4,4);  // 1st quadrant

```

```

labelX[6]=x[6]+3;
labelY[6]=y[6]-y_ADJUSTMENT;

x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(8*Math.PI/180));
y[7]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(8*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 2nd quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(22*Math.PI/180));
y[8]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(22*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 2nd quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]+y_ADJUSTMENT;

x[9]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(36*Math.PI/180));
y[9]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(36*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

x[10]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(49*Math.PI/180));
y[10]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(49*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 2nd quadrant

labelX[10]=x[10]+3;
labelY[10]=y[10]+y_ADJUSTMENT;

x[11]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(62*Math.PI/180));
y[11]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(62*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 2nd quadrant

labelX[11]=x[11]+3;
labelY[11]=y[11]+y_ADJUSTMENT;

x[12]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(75*Math.PI/180));
y[12]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(75*Math.PI/180));
g.fillOval(x[12],y[12],4,4); // 2nd quadrant

labelX[12]=x[12]+3;
labelY[12]=y[12]+y_ADJUSTMENT;

x[13]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(88*Math.PI/180));
y[13]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(88*Math.PI/180));
g.fillOval(x[13],y[13],4,4); // 2nd quadrant

labelX[13]=x[13]+3;
labelY[13]=y[13]+y_ADJUSTMENT;

x[14]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(79*Math.PI/180));
y[14]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(79*Math.PI/180));
g.fillOval(x[14],y[14],4,4); // 3rd quadrant

labelX[14]=x[14]-x_ADJUSTMENT; // move library name out of circle

```

```

labelY[14]=y[14]+y_ADJUSTMENT;    // move down

x[15]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[15]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[15],y[15],4,4);      // 3rd quadrant

labelX[15]=x[15]-x_ADJUSTMENT;    // move library name out of circle
labelY[15]=y[15]+y_ADJUSTMENT;    // move down

x[16]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(53*Math.PI/180));
y[16]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(53*Math.PI/180));
g.fillOval(x[16],y[16],4,4);      // 3rd quadrant

labelX[16]=x[16]-x_ADJUSTMENT;    // move library name out of circle
labelY[16]=y[16]+y_ADJUSTMENT;    // move down

x[17]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(40*Math.PI/180));
y[17]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(40*Math.PI/180));
g.fillOval(x[17],y[17],4,4);      // 3rd quadrant

labelX[17]=x[17]-x_ADJUSTMENT;    // move library name out of circle
labelY[17]=y[17]+y_ADJUSTMENT;    // move down

x[18]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(27*Math.PI/180));
y[18]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(27*Math.PI/180));
g.fillOval(x[18],y[18],4,4);      // 3rd quadrant

labelX[18]=x[18]-x_ADJUSTMENT;    // move library name out of circle
labelY[18]=y[18]+y_ADJUSTMENT;    // move down

x[19]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(14*Math.PI/180));
y[19]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(14*Math.PI/180));
g.fillOval(x[19],y[19],4,4);      // 3rd quadrant

labelX[19]=x[19]-x_ADJUSTMENT;    // move library name out of circle
labelY[19]=y[19]+y_ADJUSTMENT;    // move down

x[20]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(1*Math.PI/180));
y[20]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(1*Math.PI/180));
g.fillOval(x[20],y[20],4,4);      // 3rd quadrant

labelX[20]=x[20]-x_ADJUSTMENT;    // move library name out of circle
labelY[20]=y[20]+y_ADJUSTMENT;    // move down

x[21]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(12*Math.PI/180));
y[21]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(12*Math.PI/180));
g.fillOval(x[21],y[21],4,4);      // 4th quadrant

labelX[21]=x[21]-x_ADJUSTMENT;    // move library name out of circle
labelY[21]=y[21]-y_ADJUSTMENT;    // move up

x[22]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(25*Math.PI/180));
y[22]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(25*Math.PI/180));
g.fillOval(x[22],y[22],4,4);      // 4th quadrant

labelX[22]=x[22]-x_ADJUSTMENT;    // move library name out of circle
labelY[22]=y[22]-y_ADJUSTMENT;    // move up

```

```

x[23]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(38*Math.PI/180));
y[23]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(38*Math.PI/180));
g.fillOval(x[23],y[23],4,4);      // 4th quadrant

labelX[23]=x[23]-x_ADJUSTMENT;    // move library name out of circle
labelY[23]=y[23]-y_ADJUSTMENT;    // move up

x[24]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(51*Math.PI/180));
y[24]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(51*Math.PI/180));
g.fillOval(x[24],y[24],4,4);      // 4th quadrant

labelX[24]=x[24]-x_ADJUSTMENT;    // move library name out of circle
labelY[24]=y[24]-y_ADJUSTMENT;    // move up

x[25]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(64*Math.PI/180));
y[25]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(64*Math.PI/180));
g.fillOval(x[25],y[25],4,4);      // 4th quadrant

labelX[25]=x[25]-x_ADJUSTMENT;    // move library name out of circle
labelY[25]=y[25]-y_ADJUSTMENT;    // move up

x[26]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(77*Math.PI/180));
y[26]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(77*Math.PI/180));
g.fillOval(x[26],y[26],4,4);      // 4th quadrant

labelX[26]=x[26]-x_ADJUSTMENT;    // move library name out of circle
labelY[26]=y[26]-y_ADJUSTMENT;    // move up

} // end 27

else if ( 28 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4); // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

    x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(77*Math.PI/180));
    y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(77*Math.PI/180));
    g.fillOval(x[1],y[1],4,4); // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(64*Math.PI/180));
    y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(64*Math.PI/180));
    g.fillOval(x[2],y[2],4,4); // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(51*Math.PI/180));
    y[3]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(51*Math.PI/180));
    g.fillOval(x[3],y[3],4,4); // 1st quadrant

```

```

labelX[3]=x[3]+3;
labelY[3]=y[3]-y_ADJUSTMENT;

x[4]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(38*Math.PI/180));
y[4]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(38*Math.PI/180));
g.fillOval(x[4],y[4],4,4); // 1st quadrant

labelX[4]=x[4]+3;
labelY[4]=y[4]-y_ADJUSTMENT;

x[5]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(25*Math.PI/180));
y[5]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(25*Math.PI/180));
g.fillOval(x[5],y[5],4,4); // 1st quadrant

labelX[5]=x[5]+3;
labelY[5]=y[5]-y_ADJUSTMENT;

x[6]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(12*Math.PI/180));
y[6]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(12*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 1st quadrant

labelX[6]=x[6]+3;
labelY[6]=y[6]-y_ADJUSTMENT;

x[7]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(1*Math.PI/180));
y[7]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(1*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 2nd quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(14*Math.PI/180));
y[8]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(14*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 2nd quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]+y_ADJUSTMENT;

x[9]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(27*Math.PI/180));
y[9]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(27*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

x[10]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(40*Math.PI/180));
y[10]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(40*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 2nd quadrant

labelX[10]=x[10]+3;
labelY[10]=y[10]+y_ADJUSTMENT;

x[11]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(53*Math.PI/180));
y[11]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(53*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 2nd quadrant

```

```

labelX[11]=x[11]+3;
labelY[11]=y[11]+y_ADJUSTMENT;

x[12]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[12]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[12],y[12],4,4);      // 2nd quadrant

labelX[12]=x[12]+3;
labelY[12]=y[12]+y_ADJUSTMENT;

x[13]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(79*Math.PI/180));
y[13]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(79*Math.PI/180));
g.fillOval(x[13],y[13],4,4);      // 2nd quadrant

labelX[13]=x[13]+3;
labelY[13]=y[13]+y_ADJUSTMENT;

x[14]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(88*Math.PI/180));
y[14]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(88*Math.PI/180));
g.fillOval(x[14],y[14],4,4);      // 3rd quadrant

labelX[14]=x[14]-x_ADJUSTMENT;    // move library name out of circle
labelY[14]=y[14]+y_ADJUSTMENT;    // move down

x[15]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(75*Math.PI/180));
y[15]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(75*Math.PI/180));
g.fillOval(x[15],y[15],4,4);      // 3rd quadrant

labelX[15]=x[15]-x_ADJUSTMENT;    // move library name out of circle
labelY[15]=y[15]+y_ADJUSTMENT;    // move down

x[16]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(62*Math.PI/180));
y[16]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(62*Math.PI/180));
g.fillOval(x[16],y[16],4,4);      // 3rd quadrant

labelX[16]=x[16]-x_ADJUSTMENT;    // move library name out of circle
labelY[16]=y[16]+y_ADJUSTMENT;    // move down

x[17]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(49*Math.PI/180));
y[17]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(49*Math.PI/180));
g.fillOval(x[17],y[17],4,4);      // 3rd quadrant

labelX[17]=x[17]-x_ADJUSTMENT;    // move library name out of circle
labelY[17]=y[17]+y_ADJUSTMENT;    // move down

x[18]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(36*Math.PI/180));
y[18]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(36*Math.PI/180));
g.fillOval(x[18],y[18],4,4);      // 3rd quadrant

labelX[18]=x[18]-x_ADJUSTMENT;    // move library name out of circle
labelY[18]=y[18]+y_ADJUSTMENT;    // move down

x[19]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(23*Math.PI/180));
y[19]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(23*Math.PI/180));
g.fillOval(x[19],y[19],4,4);      // 3rd quadrant

labelX[19]=x[19]-x_ADJUSTMENT;    // move library name out of circle

```

```

labelY[19]=y[19]+y_ADJUSTMENT;    // move down

x[20]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(10*Math.PI/180));
y[20]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(10*Math.PI/180));
g.fillOval(x[20],y[20],4,4);      // 3rd quadrant

labelX[20]=x[20]-x_ADJUSTMENT;    // move library name out of circle
labelY[20]=y[20]+y_ADJUSTMENT;    // move down

x[21]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(3*Math.PI/180));
y[21]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(3*Math.PI/180));
g.fillOval(x[21],y[21],4,4);      // 4th quadrant

labelX[21]=x[21]-x_ADJUSTMENT;    // move library name out of circle
labelY[21]=y[21]-y_ADJUSTMENT;    // move up

x[22]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(16*Math.PI/180));
y[22]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(16*Math.PI/180));
g.fillOval(x[22],y[22],4,4);      // 4th quadrant

labelX[22]=x[22]-x_ADJUSTMENT;    // move library name out of circle
labelY[22]=y[22]-y_ADJUSTMENT;    // move up

x[23]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(29*Math.PI/180));
y[23]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(29*Math.PI/180));
g.fillOval(x[23],y[23],4,4);      // 4th quadrant

labelX[23]=x[23]-x_ADJUSTMENT;    // move library name out of circle
labelY[23]=y[23]-y_ADJUSTMENT;    // move up

x[24]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[24]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[24],y[24],4,4);      // 4th quadrant

labelX[24]=x[24]-x_ADJUSTMENT;    // move library name out of circle
labelY[24]=y[24]-y_ADJUSTMENT;    // move up

x[25]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[25]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[25],y[25],4,4);      // 4th quadrant

labelX[25]=x[25]-x_ADJUSTMENT;    // move library name out of circle
labelY[25]=y[25]-y_ADJUSTMENT;    // move up

x[26]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[26]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[26],y[26],4,4);      // 4th quadrant

labelX[26]=x[26]-x_ADJUSTMENT;    // move library name out of circle
labelY[26]=y[26]-y_ADJUSTMENT;    // move up

x[27]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[27]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[27],y[27],4,4);      // 4th quadrant

labelX[27]=x[27]-x_ADJUSTMENT;    // move library name out of circle
labelY[27]=y[27]-y_ADJUSTMENT;    // move up

```

```

    } // end 28
/** REMOVE COMMENT LATER
    else if ( 29 == numberOfLibraries)
    {
        x[0]=xCircle;
        y[0]=START_Y;
        g.fillOval(x[0],y[0],4,4); // top of circle

        labelX[0]=x[0]-x_ADJUSTMENT; // move library name out of circle
        labelY[0]=y[0]-y_ADJUSTMENT; // move up

        x[1]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(77*Math.PI/180));
        y[1]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(77*Math.PI/180));
        g.fillOval(x[1],y[1],4,4); // 1st quadrant

        labelX[1]=x[1]+3;
        labelY[1]=y[1]-y_ADJUSTMENT;

        x[2]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(64*Math.PI/180));
        y[2]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(64*Math.PI/180));
        g.fillOval(x[2],y[2],4,4); // 1st quadrant

        labelX[2]=x[2]+3;
        labelY[2]=y[2]-y_ADJUSTMENT;

        x[3]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(51*Math.PI/180));
        y[3]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(51*Math.PI/180));
        g.fillOval(x[3],y[3],4,4); // 1st quadrant

        labelX[3]=x[3]+3;
        labelY[3]=y[3]-y_ADJUSTMENT;

        x[4]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(38*Math.PI/180));
        y[4]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(38*Math.PI/180));
        g.fillOval(x[4],y[4],4,4); // 1st quadrant

        labelX[4]=x[4]+3;
        labelY[4]=y[4]-y_ADJUSTMENT;

        x[5]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(25*Math.PI/180));
        y[5]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(25*Math.PI/180));
        g.fillOval(x[5],y[5],4,4); // 1st quadrant

        labelX[5]=x[5]+3;
        labelY[5]=y[5]-y_ADJUSTMENT;

        x[6]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(12*Math.PI/180));
        y[6]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(12*Math.PI/180));
        g.fillOval(x[6],y[6],4,4); // 1st quadrant

        labelX[6]=x[6]+3;
        labelY[6]=y[6]-y_ADJUSTMENT;

        x[7]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(1*Math.PI/180));
        y[7]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(1*Math.PI/180));
        g.fillOval(x[7],y[7],4,4); // 2nd quadrant

```

```

labelX[7]=x[7]+3;
labelY[7]=y[7]+y_ADJUSTMENT;

x[8]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(14*Math.PI/180));
y[8]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(14*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 2nd quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]+y_ADJUSTMENT;

x[9]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(27*Math.PI/180));
y[9]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(27*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

x[10]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(40*Math.PI/180));
y[10]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(40*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 2nd quadrant

labelX[10]=x[10]+3;
labelY[10]=y[10]+y_ADJUSTMENT;

x[11]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(53*Math.PI/180));
y[11]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(53*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 2nd quadrant

labelX[11]=x[11]+3;
labelY[11]=y[11]+y_ADJUSTMENT;

x[12]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[12]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[12],y[12],4,4); // 2nd quadrant

labelX[12]=x[12]+3;
labelY[12]=y[12]+y_ADJUSTMENT;

x[13]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[13]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[13],y[13],4,4); // 2nd quadrant

labelX[13]=x[13]+3;
labelY[13]=y[13]+y_ADJUSTMENT;

x[14]=xCircle;
y[14]=START_Y+2*CIRCLE_RADIUS_INT;
g.fillOval(x[14],y[14],4,4); // bottom of circle

labelX[14]=x[14]-x_ADJUSTMENT; // move library name out of circle
labelY[14]=y[14]+y_ADJUSTMENT; // move down

x[15]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[15]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[15],y[15],4,4); // 3rd quadrant

```

```

labelX[15]=x[15]-x_ADJUSTMENT;    // move library name out of circle
labelY[15]=y[15]+y_ADJUSTMENT;    // move down

x[16]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[16]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[16],y[16],4,4);      // 3rd quadrant

labelX[16]=x[16]-x_ADJUSTMENT;    // move library name out of circle
labelY[16]=y[16]+y_ADJUSTMENT;    // move down

x[17]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[17]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[17],y[17],4,4);      // 3rd quadrant

labelX[17]=x[17]-x_ADJUSTMENT;    // move library name out of circle
labelY[17]=y[17]+y_ADJUSTMENT;    // move down

x[18]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[18]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[18],y[18],4,4);      // 3rd quadrant

labelX[18]=x[18]-x_ADJUSTMENT;    // move library name out of circle
labelY[18]=y[18]+y_ADJUSTMENT;    // move down

x[19]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[19]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[19],y[19],4,4);      // 3rd quadrant

labelX[19]=x[19]-x_ADJUSTMENT;    // move library name out of circle
labelY[19]=y[19]+y_ADJUSTMENT;    // move down

x[20]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[20]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[20],y[20],4,4);      // 3rd quadrant

labelX[20]=x[20]-x_ADJUSTMENT;    // move library name out of circle
labelY[20]=y[20]+y_ADJUSTMENT;    // move down

x[21]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[21]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[21],y[21],4,4);      // 3rd quadrant

labelX[21]=x[21]-x_ADJUSTMENT;    // move library name out of circle
labelY[21]=y[21]+y_ADJUSTMENT;    // move down

x[22]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[22]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[22],y[22],4,4);      // 4th quadrant

labelX[22]=x[22]-x_ADJUSTMENT;    // move library name out of circle
labelY[22]=y[22]-y_ADJUSTMENT;    // move up

x[23]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[23]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[23],y[23],4,4);      // 4th quadrant

labelX[23]=x[23]-x_ADJUSTMENT;    // move library name out of circle

```

```

    labelY[23]=y[23]-y_ADJUSTMENT;    // move up

    x[24]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
    y[24]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
    g.fillOval(x[24],y[24],4,4);    // 4th quadrant

    labelX[24]=x[24]-x_ADJUSTMENT;    // move library name out of circle
    labelY[24]=y[24]-y_ADJUSTMENT;    // move up

    x[25]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
    y[25]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
    g.fillOval(x[25],y[25],4,4);    // 4th quadrant

    labelX[25]=x[25]-x_ADJUSTMENT;    // move library name out of circle
    labelY[25]=y[25]-y_ADJUSTMENT;    // move up

    x[26]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
    y[26]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
    g.fillOval(x[26],y[26],4,4);    // 4th quadrant

    labelX[26]=x[26]-x_ADJUSTMENT;    // move library name out of circle
    labelY[26]=y[26]-y_ADJUSTMENT;    // move up

    x[27]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
    y[27]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
    g.fillOval(x[27],y[27],4,4);    // 4th quadrant

    labelX[27]=x[27]-x_ADJUSTMENT;    // move library name out of circle
    labelY[27]=y[27]-y_ADJUSTMENT;    // move up

    x[28]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
    y[28]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
    g.fillOval(x[28],y[28],4,4);    // 4th quadrant

    labelX[28]=x[28]-x_ADJUSTMENT;    // move library name out of circle
    labelY[28]=y[28]-y_ADJUSTMENT;    // move up

} // end 29

else if ( 30 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

    x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
    y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);    // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
    y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(66*Math.PI/180));

```

```

g.fillOval(x[2],y[2],4,4); // 1st quadrant

labelX[2]=x[2]+3;
labelY[2]=y[2]-y_ADJUSTMENT;

x[3]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[3]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[3],y[3],4,4); // 1st quadrant

labelX[3]=x[3]+3;
labelY[3]=y[3]-y_ADJUSTMENT;

x[4]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[4]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[4],y[4],4,4); // 1st quadrant

labelX[4]=x[4]+3;
labelY[4]=y[4]-y_ADJUSTMENT;

x[5]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[5]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[5],y[5],4,4); // 1st quadrant

labelX[5]=x[5]+3;
labelY[5]=y[5]-y_ADJUSTMENT;

x[6]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[6]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 1st quadrant

labelX[6]=x[6]+3;
labelY[6]=y[6]-y_ADJUSTMENT;

x[7]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[7]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 1st quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]-y_ADJUSTMENT;

x[8]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[8]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 2nd quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]+y_ADJUSTMENT;

x[9]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[9]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

x[10]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[10]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 2nd quadrant

```

```

labelX[10]=x[10]+3;
labelY[10]=y[10]+y_ADJUSTMENT;

x[11]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[11]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[11],y[11],4,4);      // 2nd quadrant

labelX[11]=x[11]+3;
labelY[11]=y[11]+y_ADJUSTMENT;

x[12]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[12]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[12],y[12],4,4);      // 2nd quadrant

labelX[12]=x[12]+3;
labelY[12]=y[12]+y_ADJUSTMENT;

x[13]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[13]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[13],y[13],4,4);      // 2nd quadrant

labelX[13]=x[13]+3;
labelY[13]=y[13]+y_ADJUSTMENT;

x[14]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[14]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[14],y[14],4,4);      // 2nd quadrant

labelX[14]=x[14]+3;
labelY[14]=y[14]+y_ADJUSTMENT;

x[15]=xCircle;
y[15]=START_Y+2*CIRCLE_RADIUS_INT;
g.fillOval(x[15],y[15],4,4);      // bottom of circle

labelX[15]=x[15]-x_ADJUSTMENT;    // move library name out of circle
labelY[15]=y[15]+y_ADJUSTMENT;    // move down

x[16]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[16]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[16],y[16],4,4);      // 3rd quadrant

labelX[16]=x[16]-x_ADJUSTMENT;    // move library name out of circle
labelY[16]=y[16]+y_ADJUSTMENT;    // move down

x[17]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[17]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[17],y[17],4,4);      // 3rd quadrant

labelX[17]=x[17]-x_ADJUSTMENT;    // move library name out of circle
labelY[17]=y[17]+y_ADJUSTMENT;    // move down

x[18]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[18]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[18],y[18],4,4);      // 3rd quadrant

```

```

labelX[18]=x[18]-x_ADJUSTMENT;    // move library name out of circle
labelY[18]=y[18]+y_ADJUSTMENT;    // move down

x[19]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[19]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[19],y[19],4,4);      // 3rd quadrant

labelX[19]=x[19]-x_ADJUSTMENT;    // move library name out of circle
labelY[19]=y[19]+y_ADJUSTMENT;    // move down

x[20]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[20]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[20],y[20],4,4);      // 3rd quadrant

labelX[20]=x[20]-x_ADJUSTMENT;    // move library name out of circle
labelY[20]=y[20]+y_ADJUSTMENT;    // move down

x[21]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[21]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[21],y[21],4,4);      // 3rd quadrant

labelX[21]=x[21]-x_ADJUSTMENT;    // move library name out of circle
labelY[21]=y[21]+y_ADJUSTMENT;    // move down

x[22]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[22]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[22],y[22],4,4);      // 3rd quadrant

labelX[22]=x[22]-x_ADJUSTMENT;    // move library name out of circle
labelY[22]=y[22]+y_ADJUSTMENT;    // move down

x[23]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[23]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[23],y[23],4,4);      // 4th quadrant

labelX[23]=x[23]-x_ADJUSTMENT;    // move library name out of circle
labelY[23]=y[23]-y_ADJUSTMENT;    // move up

x[24]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[24]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[24],y[24],4,4);      // 4th quadrant

labelX[24]=x[24]-x_ADJUSTMENT;    // move library name out of circle
labelY[24]=y[24]-y_ADJUSTMENT;    // move up

x[25]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[25]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[25],y[25],4,4);      // 4th quadrant

labelX[25]=x[25]-x_ADJUSTMENT;    // move library name out of circle
labelY[25]=y[25]-y_ADJUSTMENT;    // move up

x[26]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[26]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[26],y[26],4,4);      // 4th quadrant

labelX[26]=x[26]-x_ADJUSTMENT;    // move library name out of circle

```

```

labelY[26]=y[26]-y_ADJUSTMENT;    // move up

x[27]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[27]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[27],y[27],4,4);      // 4th quadrant

labelX[27]=x[27]-x_ADJUSTMENT;    // move library name out of circle
labelY[27]=y[27]-y_ADJUSTMENT;    // move up

x[28]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[28]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[28],y[28],4,4);      // 4th quadrant

labelX[28]=x[28]-x_ADJUSTMENT;    // move library name out of circle
labelY[28]=y[28]-y_ADJUSTMENT;    // move up

x[29]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[29]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[29],y[29],4,4);      // 4th quadrant

labelX[29]=x[29]-x_ADJUSTMENT;    // move library name out of circle
labelY[29]=y[29]-y_ADJUSTMENT;    // move up

} // end 30

else if ( 31 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

    x[1]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(79*Math.PI/180));
    y[1]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(79*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);    // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(68*Math.PI/180));
    y[2]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(68*Math.PI/180));
    g.fillOval(x[2],y[2],4,4);    // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(57*Math.PI/180));
    y[3]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(57*Math.PI/180));
    g.fillOval(x[3],y[3],4,4);    // 1st quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]-y_ADJUSTMENT;

    x[4]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(46*Math.PI/180));

```

```

y[4]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(46*Math.PI/180));
g.fillOval(x[4],y[4],4,4); // 1st quadrant

labelX[4]=x[4]+3;
labelY[4]=y[4]-y_ADJUSTMENT;

x[5]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(35*Math.PI/180));
y[5]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(35*Math.PI/180));
g.fillOval(x[5],y[5],4,4); // 1st quadrant

labelX[5]=x[5]+3;
labelY[5]=y[5]-y_ADJUSTMENT;

x[6]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(24*Math.PI/180));
y[6]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(24*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 1st quadrant

labelX[6]=x[6]+3;
labelY[6]=y[6]-y_ADJUSTMENT;

x[7]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(13*Math.PI/180));
y[7]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(13*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 1st quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]-y_ADJUSTMENT;

x[8]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(2*Math.PI/180));
y[8]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(2*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 1st quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]-y_ADJUSTMENT;

x[9]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(9*Math.PI/180));
y[9]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(9*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

x[10]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(20*Math.PI/180));
y[10]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(20*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 2nd quadrant

labelX[10]=x[10]+3;
labelY[10]=y[10]+y_ADJUSTMENT;

x[11]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(31*Math.PI/180));
y[11]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(31*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 2nd quadrant

labelX[11]=x[11]+3;
labelY[11]=y[11]+y_ADJUSTMENT;

x[12]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[12]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(42*Math.PI/180));

```

```

g.fillOval(x[12],y[12],4,4);          // 2nd quadrant

labelX[12]=x[12]+3;
labelY[12]=y[12]+y_ADJUSTMENT;

x[13]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[13]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[13],y[13],4,4);          // 2nd quadrant

labelX[13]=x[13]+3;
labelY[13]=y[13]+y_ADJUSTMENT;

x[14]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[14]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[14],y[14],4,4);          // 2nd quadrant

labelX[14]=x[14]+3;
labelY[14]=y[14]+y_ADJUSTMENT;

x[15]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[15]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[15],y[15],4,4);          // 2nd quadrant

labelX[15]=x[15]+3;
labelY[15]=y[15]+y_ADJUSTMENT;

x[16]=xCircle;
y[16]=START_Y+2*CIRCLE_RADIUS_INT;
g.fillOval(x[16],y[16],4,4);          // bottom of circle

labelX[16]=x[16]-x_ADJUSTMENT;        // move library name out of circle
labelY[16]=y[16]+y_ADJUSTMENT;        // move down

x[17]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[17]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[17],y[17],4,4);          // 3rd quadrant

labelX[17]=x[17]-x_ADJUSTMENT;        // move library name out of circle
labelY[17]=y[17]+y_ADJUSTMENT;        // move down

x[18]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[18]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[18],y[18],4,4);          // 3rd quadrant

labelX[18]=x[18]-x_ADJUSTMENT;        // move library name out of circle
labelY[18]=y[18]+y_ADJUSTMENT;        // move down

x[19]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[19]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[19],y[19],4,4);          // 3rd quadrant

labelX[19]=x[19]-x_ADJUSTMENT;        // move library name out of circle
labelY[19]=y[19]+y_ADJUSTMENT;        // move down

x[20]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[20]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[20],y[20],4,4);          // 3rd quadrant

```

```

labelX[20]=x[20]-x_ADJUSTMENT;    // move library name out of circle
labelY[20]=y[20]+y_ADJUSTMENT;    // move down

x[21]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[21]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[21],y[21],4,4);      // 3rd quadrant

labelX[21]=x[21]-x_ADJUSTMENT;    // move library name out of circle
labelY[21]=y[21]+y_ADJUSTMENT;    // move down

x[22]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[22]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[22],y[22],4,4);      // 3rd quadrant

labelX[22]=x[22]-x_ADJUSTMENT;    // move library name out of circle
labelY[22]=y[22]+y_ADJUSTMENT;    // move down

x[23]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[23]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[23],y[23],4,4);      // 3rd quadrant

labelX[23]=x[23]-x_ADJUSTMENT;    // move library name out of circle
labelY[23]=y[23]+y_ADJUSTMENT;    // move down

x[24]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[24]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[24],y[24],4,4);      // 4th quadrant

labelX[24]=x[24]-x_ADJUSTMENT;    // move library name out of circle
labelY[24]=y[24]-y_ADJUSTMENT;    // move up

x[25]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[25]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[25],y[25],4,4);      // 4th quadrant

labelX[25]=x[25]-x_ADJUSTMENT;    // move library name out of circle
labelY[25]=y[25]-y_ADJUSTMENT;    // move up

x[26]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[26]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[26],y[26],4,4);      // 4th quadrant

labelX[26]=x[26]-x_ADJUSTMENT;    // move library name out of circle
labelY[26]=y[26]-y_ADJUSTMENT;    // move up

x[27]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[27]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[27],y[27],4,4);      // 4th quadrant

labelX[27]=x[27]-x_ADJUSTMENT;    // move library name out of circle
labelY[27]=y[27]-y_ADJUSTMENT;    // move up

x[28]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[28]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[28],y[28],4,4);      // 4th quadrant

```

```

labelX[28]=x[28]-x_ADJUSTMENT;    // move library name out of circle
labelY[28]=y[28]-y_ADJUSTMENT;    // move up

x[29]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[29]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[29],y[29],4,4);      // 4th quadrant

labelX[29]=x[29]-x_ADJUSTMENT;    // move library name out of circle
labelY[29]=y[29]-y_ADJUSTMENT;    // move up

x[30]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[30]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[30],y[30],4,4);      // 4th quadrant

labelX[30]=x[30]-x_ADJUSTMENT;    // move library name out of circle
labelY[30]=y[30]-y_ADJUSTMENT;    // move up

} // end 31

else if ( 32 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

    x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(79*Math.PI/180));
    y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(79*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);    // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(68*Math.PI/180));
    y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(68*Math.PI/180));
    g.fillOval(x[2],y[2],4,4);    // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(57*Math.PI/180));
    y[3]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(57*Math.PI/180));
    g.fillOval(x[3],y[3],4,4);    // 1st quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]-y_ADJUSTMENT;

    x[4]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(46*Math.PI/180));
    y[4]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(46*Math.PI/180));
    g.fillOval(x[4],y[4],4,4);    // 1st quadrant

    labelX[4]=x[4]+3;
    labelY[4]=y[4]-y_ADJUSTMENT;

    x[5]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(35*Math.PI/180));

```

```

y[5]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(35*Math.PI/180));
g.fillOval(x[5],y[5],4,4); // 1st quadrant

labelX[5]=x[5]+3;
labelY[5]=y[5]-y_ADJUSTMENT;

x[6]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(24*Math.PI/180));
y[6]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(24*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 1st quadrant

labelX[6]=x[6]+3;
labelY[6]=y[6]-y_ADJUSTMENT;

x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(13*Math.PI/180));
y[7]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(13*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 1st quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]-y_ADJUSTMENT;

x[8]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(2*Math.PI/180));
y[8]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(2*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 1st quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]-y_ADJUSTMENT;

x[9]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(9*Math.PI/180));
y[9]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(9*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

x[10]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(20*Math.PI/180));
y[10]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(20*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 2nd quadrant

labelX[10]=x[10]+3;
labelY[10]=y[10]+y_ADJUSTMENT;

x[11]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(31*Math.PI/180));
y[11]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(31*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 2nd quadrant

labelX[11]=x[11]+3;
labelY[11]=y[11]+y_ADJUSTMENT;

x[12]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[12]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[12],y[12],4,4); // 2nd quadrant

labelX[12]=x[12]+3;
labelY[12]=y[12]+y_ADJUSTMENT;

x[13]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(53*Math.PI/180));
y[13]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(53*Math.PI/180));

```

```

g.fillOval(x[13],y[13],4,4);          // 2nd quadrant

labelX[13]=x[13]+3;
labelY[13]=y[13]+y_ADJUSTMENT;

x[14]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(64*Math.PI/180));
y[14]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(64*Math.PI/180));
g.fillOval(x[14],y[14],4,4);          // 2nd quadrant

labelX[14]=x[14]+3;
labelY[14]=y[14]+y_ADJUSTMENT;

x[15]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(75*Math.PI/180));
y[15]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(75*Math.PI/180));
g.fillOval(x[15],y[15],4,4);          // 2nd quadrant

labelX[15]=x[15]+3;
labelY[15]=y[15]+y_ADJUSTMENT;

x[16]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(86*Math.PI/180));
y[16]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(86*Math.PI/180));
g.fillOval(x[16],y[16],4,4);          // 2nd quadrant

labelX[16]=x[16]+3;
labelY[16]=y[16]+y_ADJUSTMENT;

x[17]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(83*Math.PI/180));
y[17]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(83*Math.PI/180));
g.fillOval(x[17],y[17],4,4);          // 3rd quadrant

labelX[17]=x[17]-x_ADJUSTMENT;        // move library name out of circle
labelY[17]=y[17]+y_ADJUSTMENT;        // move down

x[18]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(72*Math.PI/180));
y[18]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(72*Math.PI/180));
g.fillOval(x[18],y[18],4,4);          // 3rd quadrant

labelX[18]=x[18]-x_ADJUSTMENT;        // move library name out of circle
labelY[18]=y[18]+y_ADJUSTMENT;        // move down

x[19]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(61*Math.PI/180));
y[19]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(61*Math.PI/180));
g.fillOval(x[19],y[19],4,4);          // 3rd quadrant

labelX[19]=x[19]-x_ADJUSTMENT;        // move library name out of circle
labelY[19]=y[19]+y_ADJUSTMENT;        // move down

x[20]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(50*Math.PI/180));
y[20]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(50*Math.PI/180));
g.fillOval(x[20],y[20],4,4);          // 3rd quadrant

labelX[20]=x[20]-x_ADJUSTMENT;        // move library name out of circle
labelY[20]=y[20]+y_ADJUSTMENT;        // move down

x[21]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(39*Math.PI/180));
y[21]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(39*Math.PI/180));
g.fillOval(x[21],y[21],4,4);          // 3rd quadrant

```

```

labelX[21]=x[21]-x_ADJUSTMENT;    // move library name out of circle
labelY[21]=y[21]+y_ADJUSTMENT;    // move down

x[22]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(28*Math.PI/180));
y[22]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(28*Math.PI/180));
g.fillOval(x[22],y[22],4,4);      // 3rd quadrant

labelX[22]=x[22]-x_ADJUSTMENT;    // move library name out of circle
labelY[22]=y[22]+y_ADJUSTMENT;    // move down

x[23]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(17*Math.PI/180));
y[23]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(17*Math.PI/180));
g.fillOval(x[23],y[23],4,4);      // 3rd quadrant

labelX[23]=x[23]-x_ADJUSTMENT;    // move library name out of circle
labelY[23]=y[23]+y_ADJUSTMENT;    // move down

x[24]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[24]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[24],y[24],4,4);      // 3rd quadrant

labelX[24]=x[24]-x_ADJUSTMENT;    // move library name out of circle
labelY[24]=y[24]+y_ADJUSTMENT;    // move down

x[25]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[25]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[25],y[25],4,4);      // 4th quadrant

labelX[25]=x[25]-x_ADJUSTMENT;    // move library name out of circle
labelY[25]=y[25]-y_ADJUSTMENT;    // move up

x[26]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(18*Math.PI/180));
y[26]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(18*Math.PI/180));
g.fillOval(x[26],y[26],4,4);      // 4th quadrant

labelX[26]=x[26]-x_ADJUSTMENT;    // move library name out of circle
labelY[26]=y[26]-y_ADJUSTMENT;    // move up

x[27]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[27]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[27],y[27],4,4);      // 4th quadrant

labelX[27]=x[27]-x_ADJUSTMENT;    // move library name out of circle
labelY[27]=y[27]-y_ADJUSTMENT;    // move up

x[28]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[28]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[28],y[28],4,4);      // 4th quadrant

labelX[28]=x[28]-x_ADJUSTMENT;    // move library name out of circle
labelY[28]=y[28]-y_ADJUSTMENT;    // move up

x[29]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(54*Math.PI/180));
y[29]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(54*Math.PI/180));
g.fillOval(x[29],y[29],4,4);      // 4th quadrant

```

```

labelX[29]=x[29]-x_ADJUSTMENT;    // move library name out of circle
labelY[29]=y[29]-y_ADJUSTMENT;    // move up

x[30]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(66*Math.PI/180));
y[30]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(66*Math.PI/180));
g.fillOval(x[30],y[30],4,4);      // 4th quadrant

labelX[30]=x[30]-x_ADJUSTMENT;    // move library name out of circle
labelY[30]=y[30]-y_ADJUSTMENT;    // move up

x[31]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(78*Math.PI/180));
y[31]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(78*Math.PI/180));
g.fillOval(x[31],y[31],4,4);      // 4th quadrant

labelX[31]=x[31]-x_ADJUSTMENT;    // move library name out of circle
labelY[31]=y[31]-y_ADJUSTMENT;    // move up

} // end 32

else if ( 33 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

    x[1]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(79*Math.PI/180));
    y[1]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(79*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);    // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(68*Math.PI/180));
    y[2]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(68*Math.PI/180));
    g.fillOval(x[2],y[2],4,4);    // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(57*Math.PI/180));
    y[3]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(57*Math.PI/180));
    g.fillOval(x[3],y[3],4,4);    // 1st quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]-y_ADJUSTMENT;

    x[4]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(46*Math.PI/180));
    y[4]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(46*Math.PI/180));
    g.fillOval(x[4],y[4],4,4);    // 1st quadrant

    labelX[4]=x[4]+3;
    labelY[4]=y[4]-y_ADJUSTMENT;

    x[5]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(35*Math.PI/180));

```

```

y[5]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(35*Math.PI/180));
g.fillOval(x[5],y[5],4,4); // 1st quadrant

labelX[5]=x[5]+3;
labelY[5]=y[5]-y_ADJUSTMENT;

x[6]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(24*Math.PI/180));
y[6]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(24*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 1st quadrant

labelX[6]=x[6]+3;
labelY[6]=y[6]-y_ADJUSTMENT;

x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(13*Math.PI/180));
y[7]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(13*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 1st quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]-y_ADJUSTMENT;

x[8]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(2*Math.PI/180));
y[8]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(2*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 1st quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]-y_ADJUSTMENT;

x[9]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(9*Math.PI/180));
y[9]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(9*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

x[10]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(20*Math.PI/180));
y[10]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(20*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 2nd quadrant

labelX[10]=x[10]+3;
labelY[10]=y[10]+y_ADJUSTMENT;

x[11]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(31*Math.PI/180));
y[11]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(31*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 2nd quadrant

labelX[11]=x[11]+3;
labelY[11]=y[11]+y_ADJUSTMENT;

x[12]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[12]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(42*Math.PI/180));
g.fillOval(x[12],y[12],4,4); // 2nd quadrant

labelX[12]=x[12]+3;
labelY[12]=y[12]+y_ADJUSTMENT;

x[13]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(53*Math.PI/180));
y[13]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(53*Math.PI/180));

```

```

g.fillOval(x[13],y[13],4,4);          // 2nd quadrant

labelX[13]=x[13]+3;
labelY[13]=y[13]+y_ADJUSTMENT;

x[14]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(64*Math.PI/180));
y[14]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(64*Math.PI/180));
g.fillOval(x[14],y[14],4,4);          // 2nd quadrant

labelX[14]=x[14]+3;
labelY[14]=y[14]+y_ADJUSTMENT;

x[15]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(75*Math.PI/180));
y[15]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(75*Math.PI/180));
g.fillOval(x[15],y[15],4,4);          // 2nd quadrant

labelX[15]=x[15]+3;
labelY[15]=y[15]+y_ADJUSTMENT;

x[16]=xCircle+(int)(CIRCLE_RADIUS*Math.cos(86*Math.PI/180));
y[16]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(86*Math.PI/180));
g.fillOval(x[16],y[16],4,4);          // 2nd quadrant

labelX[16]=x[16]+3;
labelY[16]=y[16]+y_ADJUSTMENT;

x[17]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(83*Math.PI/180));
y[17]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(83*Math.PI/180));
g.fillOval(x[17],y[17],4,4);          // 3rd quadrant

labelX[17]=x[17]-x_ADJUSTMENT;        // move library name out of circle
labelY[17]=y[17]+y_ADJUSTMENT;        // move down

x[18]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(72*Math.PI/180));
y[18]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(72*Math.PI/180));
g.fillOval(x[18],y[18],4,4);          // 3rd quadrant

labelX[18]=x[18]-x_ADJUSTMENT;        // move library name out of circle
labelY[18]=y[18]+y_ADJUSTMENT;        // move down

x[19]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(61*Math.PI/180));
y[19]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(61*Math.PI/180));
g.fillOval(x[19],y[19],4,4);          // 3rd quadrant

labelX[19]=x[19]-x_ADJUSTMENT;        // move library name out of circle
labelY[19]=y[19]+y_ADJUSTMENT;        // move down

x[20]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(50*Math.PI/180));
y[20]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(50*Math.PI/180));
g.fillOval(x[20],y[20],4,4);          // 3rd quadrant

labelX[20]=x[20]-x_ADJUSTMENT;        // move library name out of circle
labelY[20]=y[20]+y_ADJUSTMENT;        // move down

x[21]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(39*Math.PI/180));
y[21]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(39*Math.PI/180));
g.fillOval(x[21],y[21],4,4);          // 3rd quadrant

```

```

labelX[21]=x[21]-x_ADJUSTMENT;    // move library name out of circle
labelY[21]=y[21]+y_ADJUSTMENT;    // move down

x[22]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(28*Math.PI/180));
y[22]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(28*Math.PI/180));
g.fillOval(x[22],y[22],4,4);      // 3rd quadrant

labelX[22]=x[22]-x_ADJUSTMENT;    // move library name out of circle
labelY[22]=y[22]+y_ADJUSTMENT;    // move down

x[23]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(17*Math.PI/180));
y[23]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(17*Math.PI/180));
g.fillOval(x[23],y[23],4,4);      // 3rd quadrant

labelX[23]=x[23]-x_ADJUSTMENT;    // move library name out of circle
labelY[23]=y[23]+y_ADJUSTMENT;    // move down

x[24]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(6*Math.PI/180));
y[24]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(6*Math.PI/180));
g.fillOval(x[24],y[24],4,4);      // 3rd quadrant

labelX[24]=x[24]-x_ADJUSTMENT;    // move library name out of circle
labelY[24]=y[24]+y_ADJUSTMENT;    // move down

x[25]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(5*Math.PI/180));
y[25]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(5*Math.PI/180));
g.fillOval(x[25],y[25],4,4);      // 4th quadrant

labelX[25]=x[25]-x_ADJUSTMENT;    // move library name out of circle
labelY[25]=y[25]-y_ADJUSTMENT;    // move up

x[26]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(16*Math.PI/180));
y[26]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(16*Math.PI/180));
g.fillOval(x[26],y[26],4,4);      // 4th quadrant

labelX[26]=x[26]-x_ADJUSTMENT;    // move library name out of circle
labelY[26]=y[26]-y_ADJUSTMENT;    // move up

x[27]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(27*Math.PI/180));
y[27]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(27*Math.PI/180));
g.fillOval(x[27],y[27],4,4);      // 4th quadrant

labelX[27]=x[27]-x_ADJUSTMENT;    // move library name out of circle
labelY[27]=y[27]-y_ADJUSTMENT;    // move up

x[28]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(38*Math.PI/180));
y[28]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(38*Math.PI/180));
g.fillOval(x[28],y[28],4,4);      // 4th quadrant

labelX[28]=x[28]-x_ADJUSTMENT;    // move library name out of circle
labelY[28]=y[28]-y_ADJUSTMENT;    // move up

x[29]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(49*Math.PI/180));
y[29]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(49*Math.PI/180));
g.fillOval(x[29],y[29],4,4);      // 4th quadrant

```

```

labelX[29]=x[29]-x_ADJUSTMENT;    // move library name out of circle
labelY[29]=y[29]-y_ADJUSTMENT;    // move up

x[30]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
y[30]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
g.fillOval(x[30],y[30],4,4);      // 4th quadrant

labelX[30]=x[30]-x_ADJUSTMENT;    // move library name out of circle
labelY[30]=y[30]-y_ADJUSTMENT;    // move up

x[31]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(70*Math.PI/180));
y[31]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(70*Math.PI/180));
g.fillOval(x[31],y[31],4,4);      // 4th quadrant

labelX[31]=x[31]-x_ADJUSTMENT;    // move library name out of circle
labelY[31]=y[31]-y_ADJUSTMENT;    // move up

x[32]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(80*Math.PI/180));
y[32]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(80*Math.PI/180));
g.fillOval(x[32],y[32],4,4);      // 4th quadrant

labelX[32]=x[32]-x_ADJUSTMENT;    // move library name out of circle
labelY[32]=y[32]-y_ADJUSTMENT;    // move up

} // end 33

else if ( 34 == numberOfLibraries)
{
    x[0]=xCircle;
    y[0]=START_Y;
    g.fillOval(x[0],y[0],4,4);    // top of circle

    labelX[0]=x[0]-x_ADJUSTMENT;    // move library name out of circle
    labelY[0]=y[0]-y_ADJUSTMENT;    // move up

    x[1]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(79*Math.PI/180));
    y[1]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(79*Math.PI/180));
    g.fillOval(x[1],y[1],4,4);    // 1st quadrant

    labelX[1]=x[1]+3;
    labelY[1]=y[1]-y_ADJUSTMENT;

    x[2]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(68*Math.PI/180));
    y[2]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(68*Math.PI/180));
    g.fillOval(x[2],y[2],4,4);    // 1st quadrant

    labelX[2]=x[2]+3;
    labelY[2]=y[2]-y_ADJUSTMENT;

    x[3]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(57*Math.PI/180));
    y[3]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(57*Math.PI/180));
    g.fillOval(x[3],y[3],4,4);    // 1st quadrant

    labelX[3]=x[3]+3;
    labelY[3]=y[3]-y_ADJUSTMENT;

    x[4]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(46*Math.PI/180));

```

```

y[4]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(46*Math.PI/180));
g.fillOval(x[4],y[4],4,4); // 1st quadrant

labelX[4]=x[4]+3;
labelY[4]=y[4]-y_ADJUSTMENT;

x[5]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(35*Math.PI/180));
y[5]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(35*Math.PI/180));
g.fillOval(x[5],y[5],4,4); // 1st quadrant

labelX[5]=x[5]+3;
labelY[5]=y[5]-y_ADJUSTMENT;

x[6]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(24*Math.PI/180));
y[6]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(24*Math.PI/180));
g.fillOval(x[6],y[6],4,4); // 1st quadrant

labelX[6]=x[6]+3;
labelY[6]=y[6]-y_ADJUSTMENT;

x[7]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(13*Math.PI/180));
y[7]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(13*Math.PI/180));
g.fillOval(x[7],y[7],4,4); // 1st quadrant

labelX[7]=x[7]+3;
labelY[7]=y[7]-y_ADJUSTMENT;

x[8]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(2*Math.PI/180));
y[8]=yCircle-(int) (CIRCLE_RADIUS*Math.sin(2*Math.PI/180));
g.fillOval(x[8],y[8],4,4); // 1st quadrant

labelX[8]=x[8]+3;
labelY[8]=y[8]-y_ADJUSTMENT;

x[9]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(9*Math.PI/180));
y[9]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(9*Math.PI/180));
g.fillOval(x[9],y[9],4,4); // 2nd quadrant

labelX[9]=x[9]+3;
labelY[9]=y[9]+y_ADJUSTMENT;

x[10]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(20*Math.PI/180));
y[10]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(20*Math.PI/180));
g.fillOval(x[10],y[10],4,4); // 2nd quadrant

labelX[10]=x[10]+3;
labelY[10]=y[10]+y_ADJUSTMENT;

x[11]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(31*Math.PI/180));
y[11]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(31*Math.PI/180));
g.fillOval(x[11],y[11],4,4); // 2nd quadrant

labelX[11]=x[11]+3;
labelY[11]=y[11]+y_ADJUSTMENT;

x[12]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(42*Math.PI/180));
y[12]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(42*Math.PI/180));

```

```

    g.fillOval(x[12],y[12],4,4);          // 2nd quadrant

    labelX[12]=x[12]+3;
    labelY[12]=y[12]+y_ADJUSTMENT;

    x[13]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(53*Math.PI/180));
    y[13]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(53*Math.PI/180));
    g.fillOval(x[13],y[13],4,4);          // 2nd quadrant

    labelX[13]=x[13]+3;
    labelY[13]=y[13]+y_ADJUSTMENT;

    x[14]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(64*Math.PI/180));
    y[14]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(64*Math.PI/180));
    g.fillOval(x[14],y[14],4,4);          // 2nd quadrant

    labelX[14]=x[14]+3;
    labelY[14]=y[14]+y_ADJUSTMENT;

    x[15]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(75*Math.PI/180));
    y[15]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(75*Math.PI/180));
    g.fillOval(x[15],y[15],4,4);          // 2nd quadrant

    labelX[15]=x[15]+3;
    labelY[15]=y[15]+y_ADJUSTMENT;

    x[16]=xCircle+(int) (CIRCLE_RADIUS*Math.cos(86*Math.PI/180));
    y[16]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(86*Math.PI/180));
    g.fillOval(x[16],y[16],4,4);          // 2nd quadrant

    labelX[16]=x[16]+3;
    labelY[16]=y[16]+y_ADJUSTMENT;

    x[17]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(83*Math.PI/180));
    y[17]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(83*Math.PI/180));
    g.fillOval(x[17],y[17],4,4);          // 3rd quadrant

    labelX[17]=x[17]-x_ADJUSTMENT;          // move library name out of circle
    labelY[17]=y[17]+y_ADJUSTMENT;          // move down

    x[18]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(72*Math.PI/180));
    y[18]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(72*Math.PI/180));
    g.fillOval(x[18],y[18],4,4);          // 3rd quadrant

    labelX[18]=x[18]-x_ADJUSTMENT;          // move library name out of circle
    labelY[18]=y[18]+y_ADJUSTMENT;          // move down

    x[19]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(61*Math.PI/180));
    y[19]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(61*Math.PI/180));
    g.fillOval(x[19],y[19],4,4);          // 3rd quadrant

    labelX[19]=x[19]-x_ADJUSTMENT;          // move library name out of circle
    labelY[19]=y[19]+y_ADJUSTMENT;          // move down

    x[20]=xCircle-(int) (CIRCLE_RADIUS*Math.cos(50*Math.PI/180));
    y[20]=yCircle+(int) (CIRCLE_RADIUS*Math.sin(50*Math.PI/180));
    g.fillOval(x[20],y[20],4,4);          // 3rd quadrant

```

```

labelX[20]=x[20]-x_ADJUSTMENT;    // move library name out of circle
labelY[20]=y[20]+y_ADJUSTMENT;    // move down

x[21]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(40*Math.PI/180));
y[21]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(40*Math.PI/180));
g.fillOval(x[21],y[21],4,4);      // 3rd quadrant

labelX[21]=x[21]-x_ADJUSTMENT;    // move library name out of circle
labelY[21]=y[21]+y_ADJUSTMENT;    // move down

x[22]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[22]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[22],y[22],4,4);      // 3rd quadrant

labelX[22]=x[22]-x_ADJUSTMENT;    // move library name out of circle
labelY[22]=y[22]+y_ADJUSTMENT;    // move down

x[23]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(20*Math.PI/180));
y[23]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(20*Math.PI/180));
g.fillOval(x[23],y[23],4,4);      // 3rd quadrant

labelX[23]=x[23]-x_ADJUSTMENT;    // move library name out of circle
labelY[23]=y[23]+y_ADJUSTMENT;    // move down

x[24]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(10*Math.PI/180));
y[24]=yCircle+(int)(CIRCLE_RADIUS*Math.sin(10*Math.PI/180));
g.fillOval(x[24],y[24],4,4);      // 3rd quadrant

labelX[24]=x[24]-x_ADJUSTMENT;    // move library name out of circle
labelY[24]=y[24]+y_ADJUSTMENT;    // move down

x[25]=START_X;
y[25]=yCircle;
g.fillOval(x[25],y[25],4,4);      // left

labelX[25]=x[25]-x_ADJUSTMENT;    // move library name out of circle
labelY[25]=y[25]+y_ADJUSTMENT;    // move down

x[26]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(10*Math.PI/180));
y[26]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(10*Math.PI/180));
g.fillOval(x[26],y[26],4,4);      // 4th quadrant

labelX[26]=x[26]-x_ADJUSTMENT;    // move library name out of circle
labelY[26]=y[26]-y_ADJUSTMENT;    // move up

x[27]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(20*Math.PI/180));
y[27]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(20*Math.PI/180));
g.fillOval(x[27],y[27],4,4);      // 4th quadrant

labelX[27]=x[27]-x_ADJUSTMENT;    // move library name out of circle
labelY[27]=y[27]-y_ADJUSTMENT;    // move up

x[28]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(30*Math.PI/180));
y[28]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(30*Math.PI/180));
g.fillOval(x[28],y[28],4,4);      // 4th quadrant

```

```

labelX[28]=x[28]-x_ADJUSTMENT;    // move library name out of circle
labelY[28]=y[28]-y_ADJUSTMENT;    // move up

x[29]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(40*Math.PI/180));
y[29]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(40*Math.PI/180));
g.fillOval(x[29],y[29],4,4);      // 4th quadrant

labelX[29]=x[29]-x_ADJUSTMENT;    // move library name out of circle
labelY[29]=y[29]-y_ADJUSTMENT;    // move up

x[30]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(50*Math.PI/180));
y[30]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(50*Math.PI/180));
g.fillOval(x[30],y[30],4,4);      // 4th quadrant

labelX[30]=x[30]-x_ADJUSTMENT;    // move library name out of circle
labelY[30]=y[30]-y_ADJUSTMENT;    // move up

x[31]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(60*Math.PI/180));
y[31]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(60*Math.PI/180));
g.fillOval(x[31],y[31],4,4);      // 4th quadrant

labelX[31]=x[31]-x_ADJUSTMENT;    // move library name out of circle
labelY[31]=y[31]-y_ADJUSTMENT;    // move up

x[32]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(70*Math.PI/180));
y[32]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(70*Math.PI/180));
g.fillOval(x[32],y[32],4,4);      // 4th quadrant

labelX[32]=x[32]-x_ADJUSTMENT;    // move library name out of circle
labelY[32]=y[32]-y_ADJUSTMENT;    // move up

x[33]=xCircle-(int)(CIRCLE_RADIUS*Math.cos(80*Math.PI/180));
y[33]=yCircle-(int)(CIRCLE_RADIUS*Math.sin(80*Math.PI/180));
g.fillOval(x[33],y[33],4,4);      // 4th quadrant

labelX[33]=x[33]-x_ADJUSTMENT;    // move library name out of circle
labelY[33]=y[33]-y_ADJUSTMENT;    // move up

} // end 34
*/

// place library names next to dots
int cutoff=0;                      // to trim any overly-long sort feature
strings

for (int i =0; i < numberOfLibraries; i++)
{
    g.setColor(DARK_BROWN);        // reset color for library name labels
    g.drawString(""+sortedLibraryNames[i].trim(),labelX[i],labelY[i]); //
    trim() to remove \r spcl char

    // put sort category under library label
    g.setFont(tinyFont);            // need smaller font for longer sort
    features

    if ( sortAlready.equals("custom") ) // initial or custom sort
    {

```

```

        cutoff = sortedCustom[i].length() <= MAX_LENGTH ?
sortedCustom[i].length() : MAX_LENGTH;

g.drawString(""+sortedCustom[i].substring(0,cutoff),labelX[i],labelY[i]+8);
    }
    else if ( sortAlready.equals("germplasm") )
    {
        // use length of sort feature string if less than or equal to 'cutoff'
        cutoff = sortedGermplasm[i].length() <= MAX_LENGTH ?
sortedGermplasm[i].length() : MAX_LENGTH;

g.drawString(""+sortedGermplasm[i].substring(0,cutoff),labelX[i],labelY[i]+8);
    }
    else if ( sortAlready.equals("tissue") )
    {
        cutoff = sortedTissue[i].length() <= MAX_LENGTH ?
sortedTissue[i].length() : MAX_LENGTH;

g.drawString(""+sortedTissue[i].substring(0,cutoff),labelX[i],labelY[i]+8);
    }
    else if ( sortAlready.equals("dev_stage") )
    {
        cutoff = sortedStage[i].length() <= MAX_LENGTH ? sortedStage[i].length()
: MAX_LENGTH;

g.drawString(""+sortedStage[i].substring(0,cutoff),labelX[i],labelY[i]+8);
    }
    else if ( sortAlready.equals("condition") )
    {
        cutoff = sortedCondition[i].length() <= MAX_LENGTH ?
sortedCondition[i].length() : MAX_LENGTH;

g.drawString(""+sortedCondition[i].substring(0,cutoff),labelX[i],labelY[i]+8);
    }
    else if ( sortAlready.equals("species") )
    {
        cutoff = sortedSpecies[i].length() <= MAX_LENGTH ?
sortedSpecies[i].length() : MAX_LENGTH;

g.drawString(""+sortedSpecies[i].substring(0,cutoff),labelX[i],labelY[i]+8);
    }

    g.setFont(smallFont);    // back to font for library names

    // color bars
    g.setColor((Color)(sortColor[sortColorIndex[i]])); // e.g., index is 2 for
3rd sort feature
    g.fillRect(labelX[i],labelY[i]+9,50,2); // colored bar under sort
feature label (2 pixels wide so that contigs not covered)
    // works, but too short for some categories:
//g.fillRect(x[i],y[i]+13,cutoff*4,2);

    } // end for
}
catch (Exception e)
{
    errl+="drawLib error: "+e;

```

```

    }

} // end drawLibraries() // takes virtually no time
//*****

String findContigForXY(int mouseX, int mouseY) // called by mouseMoved() &
mouseClicked()
//*****

// find contig for mouseclick/mouseover
//
// NEEDS TO BE SYNCHRONIZED?
// probably not, as click and mouseover events already wait for each other
// to finish first (and therefore wouldn't collide at findContigForXY())
{
    int i = 0;
    boolean contigNotFound = true;
    String selectedContig="";

    // match against x,y coordinates in xCoordContig to determine contig
    try
    {
        if (showEqual) // EQUAL (equal) set of contigs
        {
            while ( (contigNotFound) && (i < numberOfContigs) )// keep on looking
for contig
            {
                if ( ( mouseX == xCoordEContig[i] )
                    && ( mouseY == yCoordEContig[i] ) )
                {
                    contigNotFound = false;
                    selectedContig = contigNames[i];
                }
                i++;
            } // end while contigNotFound
        }
        else if (showProportional) // look at PROPORTIONAL set of contigs
        {
            while ( (contigNotFound) && (i < numberOfContigs) )// keep on looking
for contig
            {
                if ( ( mouseX == xCoordPContig[i] )
                    && ( mouseY == yCoordPContig[i] ) )
                {
                    contigNotFound = false;
                    selectedContig = contigNames[i];
                }
                i++;
            } // end while contigNotFound
        }
        else // WEIGHTED set of contigs
        {
            while ( (contigNotFound) && (i < numberOfContigs) )// keep on looking
for contig
            {
                if ( ( mouseX == xCoordWContig[i] )
                    && ( mouseY == yCoordWContig[i] ) )

```

```

        {
            contigNotFound = false;
            selectedContig = contigNames[i];
        }
        i++;
    } // end while contigNotFound
} // end if showEqual
}
catch (Exception e)
{
    err1+="findContigForXY(): "+e;
}
return selectedContig;
} // end findContigForXY()
//*****
void addToContigDataString() // if user entered more than one contig in
textField
//*****

{
    String requestedContig;
    int requestedContigSize;
    String libraryInContig;
    String est[];

    allContigsFound=true; // assume this is true initially (for
errmsg)

    try
    {
        contigDataString=""; // to avoid repeating info on first contig

        for (int i=1; i < numInContigList; i++) // cycle from 2nd thru final
contig in list
        {
            if ( iContigs[i] != -1 ) // contig found
            {
                requestedContig = contigNames[iContigs[i]];
                contigDataString=contigDataString+requestedContig+": ";

                // find libraries for this contig
                estObjs = server.getESTNumberOfLibrary(requestedContig);
                requestedContigSize = estObjs.length; // # libraries in this contig

                if (requestedContigSize > 1)
                {
                    contigDataString = contigDataString + requestedContigSize+"
libraries\n";
                }
                else if (requestedContigSize == 1)
                {
                    contigDataString = contigDataString + " 1 library\n";
                }

                // list libraries and ESTs comprising libraries
                for (int j=0; j < requestedContigSize; j++) // loop thru libraries in
contig

```

```

        {
            libraryInContig = estObjs[j].getLibName();           // name of each
library
            est = server.getESTNames(requestedContig,libraryInContig); // get
array of ESTs in this contig/library

            for (int k=0; k< est.length; k++)
            {
                contigDataString = contigDataString +
libraryInContig.trim()+"\t"+est[k]+"\\n";
            }
        } // end for requestedContig's component libraries
    }
    else if (allContigsFound) // iContigs[i] == -1 && allContigsFound
still as initially set
    {
        allContigsFound=false;
    } // end if contig found
    } // end for numInContigList
}
catch (Exception e)
{
    err1+="addToContigDataString() error: "+e;
}
} // end addToContigDataString()
//*****
void getClickedContigData() // to build contigDataString if no lib lines
needed
//*****

{
    int clickedContigSize;
    String libraryInContig;
    String est[];

    try
    {
        contigDataString=clickedContig+": ";

        // find libraries for this contig
        estObjs = server.getESTNumberOfLibrary(clickedContig);
        clickedContigSize = estObjs.length; // # libraries in this contig

        if (clickedContigSize > 1)
        {
            contigDataString = contigDataString + clickedContigSize+"
libraries\\n";
        }
        else if (clickedContigSize == 1)
        {
            contigDataString = contigDataString + " 1 library\\n";
        }

        // list libraries and ESTs comprising libraries
        for (int j=0; j < clickedContigSize; j++) // loop thru libraries in
contig
        {

```

```

        libraryInContig = estObjs[j].getLibName();           // name of each
library
        est = server.getESTNames(clickedContig,libraryInContig); // get
array of ESTs in this contig/library

        for (int k=0; k< est.length; k++)
        {
            contigDataString = contigDataString +
libraryInContig.trim()+"\t"+est[k]+"\\n";
        }
    } // end for clickedContig's component libraries

    } catch (Exception e)
    {
        err1+="getClickedContigData() error: "+e;
    }
} // end getClickedContigData()
//*****

void drawLibraryLines(Graphics g, int xCoord, int yCoord, String
requestedContig)
// draw lines out from contig to contributing libraries
//*****

{
    int requestedContigSize;
    String libraryInContig;
    int numESTs;
    boolean libraryNotFound = true;
    String est[];

    g.setColor(Color.black);

    if (lookingForContig) // user selected contig(s) by using textbox
    {
        if (showEqual)
        {
            g.drawString(""+requestedContig+" (E)",xCoord,yCoord); // label the
contig
        }
        else if (showProportional)
        {
            g.drawString(""+requestedContig+" (P)",xCoord,yCoord); // label the
contig
        }
        else // showWeighted
        {
            g.drawString(""+requestedContig+" (W)",xCoord,yCoord); // label the
contig
        }
    }
    else // user selected contig by clicking
    {
        g.drawString(""+requestedContig+clickedMode,clickMouseX,clickMouseY);
    } // end if
    contigDataString=requestedContig+": ";

```

```

// find libraries for this contig
try
{
    estObjs = server.getESTNumberOfLibrary(requestedContig);
    requestedContigSize = estObjs.length;    // # libraries in this contig,
used for "eSizeOfMovement"

    if (requestedContigSize > 1)
    {
        contigDataString=contigDataString+requestedContigSize+" libraries\n";
    }
    else if (requestedContigSize == 1)
    {
        contigDataString=contigDataString+" 1 library\n";
    }
}

// ***LATER: compare whether better to use contigs or libraryNames as outer loop
// outer loop
for (int j=0; j < requestedContigSize; j++)    // loop thru libraries in
contig
{
    libraryInContig = estObjs[j].getLibName();    // name of library
    numESTs = estObjs[j].getNumOfEsts();    // # ESTs from that
library

    // find x,y coordinates for associated library using x[],y[]
    libraryNotFound = true;
    int k=0;

    // inner loop
    while ( (libraryNotFound) && (k < numberOfLibraries) )// keep on trying
to match library in contig against library on circle
    {
        if ( sortedLibraryNames[k].equals(libraryInContig) )
        {
            libraryNotFound = false;
            int xLib=x[k];
            int yLib=y[k];

            // draw lines from contig to associated library

            g.setColor(Color.red);    // DARK_RED doesn't stand out enough
            g.drawLine(xCoord,yCoord,xLib,yLib);

            // draw number of ESTs from library along library lines
            if ( (showProportional) || (showWeighted) )
            {
                if (xCoord > xLib)    // contig to right of library
                {
                    if (yCoord > yLib)    // contig position vertically under library
                    {
                        g.drawString(""+numESTs,xLib+((xCoord-xLib)/2),yLib+((yCoord-
yLib)/2));
                    }
                    else // yCoord <= yLib    (contig position vertically above
library)
                    {

```

```

        g.drawString(""+numESTs,xLib+((xCoord-xLib)/2),yCoord+((yLib-
yCoord)/2));
    }
    }
    else // xCoord <= xLib (contig to left of library)
    {
        if (yCoord > yLib)
        {
            g.drawString(""+numESTs,xCoord+((xLib-xCoord)/2),yLib+((yCoord-
yLib)/2));
        }
        else // yCoord <= yLib
        {
            g.drawString(""+numESTs,xCoord+((xLib-xCoord)/2),yCoord+((yLib-
yCoord)/2));
        }
    } // end if xCoord > xLib
} // end if showProportional

// add color and #ESTs to label for associated library

if ( (showProportional) || (showEqual) )
{
    g.drawString(""+libraryInContig.trim()+"
("+numESTs+")",labelX[k],labelY[k]);
}
else // showWeighted -- label with total # ESTs
{
    g.drawString(""+libraryInContig.trim()+"
("+estCountBySortedLib[k]+")",labelX[k],labelY[k]);
}
} // end if library.equals(libraryInContig)

k++; // go to next library in libraryNames array
} // end while libraryNotFound

// build string for whatsInContig textarea
est = server.getESTNames(requestedContig,libraryInContig);
for (int i=0; i< est.length; i++)
{
    contigDataString=contigDataString+libraryInContig.trim()+"\t"+est[i]+"
";
}
// end build string for whatsInContig textarea

} // end for requestedContig's component libraries
}
catch (Exception e)
{
    err1+="drawLibraryLines error: "+e;
}
} // end drawLibraryLines()
//*****

public void sortBySpecies() // if "species" in Choice List picked
    // sort libraries based on species

```

```

//*****
{
System.arraycopy(libraryNamesBySpecies,0,sortedLibraryNames,0,numberOfLibraries)
;

System.arraycopy(estCountBySpecies,0,estCountBySortedLib,0,numberOfLibraries);

    // create string for text area and
    // establish colors for library names
    librarySortString = "Sort by species:\n";

    for (int j = 0; j < numberOfLibraries; j++)
    {
        if ( (j==0) ||
            ( !sortedSpecies[j].equals(sortedSpecies[j-1]) )           // start new
row
        )
        {
            librarySortString = librarySortString + "\n" + sortedSpecies[j] + ": "
                + sortedLibraryNames[j].trim();
            colorCounter++;           // change to new color
        }
        else // library from same species
        {
            librarySortString = librarySortString + ",
"+sortedLibraryNames[j].trim();
        }
        // species same->same color; species different->different color
        sortColorIndex[j]=(colorCounter % MAX_NO_COLORS);           // in case # species
> # colors
    } // end for j
} // end sortBySpecies()
//*****

    public void sortByGermplasm()           // if "germplasm" in Choice List picked
        // sort libraries based on germplasm
//*****

    {
        // copy libraryNamesByGermplasm[] to standard sortedLibraryName[]

System.arraycopy(libraryNamesByGermplasm,0,sortedLibraryNames,0,numberOfLibrarie
s);

System.arraycopy(estCountByGermplasm,0,estCountBySortedLib,0,numberOfLibraries);

    // create string for text area
    // set library sort colors
    librarySortString = "Sort by germplasm:\n";

    for (int j = 0; j < numberOfLibraries; j++)
    {
        if ( (j==0) ||
            ( !sortedGermplasm[j].equals(sortedGermplasm[j-1]) ) )
        )

```

```

        {
            librarySortString = librarySortString + "\n" + sortedGermplasm[j] + ": "
                + sortedLibraryNames[j].trim();
            colorCounter++;          // change to new color
        }
        else // library from same germplasm
        {
            librarySortString = librarySortString + ",
"+sortedLibraryNames[j].trim();
        }
        // germplasm same->same color; germplasm different->different color
        sortColorIndex[j]=(colorCounter % MAX_NO_COLORS);    // in case #
        germplasm > # colors
    }
} // end sortByGermplasm()
//*****

public void sortByTissue() // if "tissue" in Choice List picked
    // sort libraries based on tissue
//*****

{
    System.arraycopy(libraryNamesByTissue,0,sortedLibraryNames,0,numberOfLibraries);
    System.arraycopy(estCountByTissue,0,estCountBySortedLib,0,numberOfLibraries);

    // create string for text area
    // establish colors for library names
    librarySortString = "Sort by tissue:\n";

    for (int j = 0; j < numberOfLibraries; j++)
    {
        if ( (j==0) ||
            ( !sortedTissue[j].equals(sortedTissue[j-1]) ) )
        {
            librarySortString = librarySortString + "\n" + sortedTissue[j] + ": "
                + sortedLibraryNames[j].trim();
            colorCounter++;          // change to new color
        }
        else // library from same tissue
        {
            librarySortString = librarySortString + ",
"+sortedLibraryNames[j].trim();
        }
        // tissue same->same color; tissue different->different color
        sortColorIndex[j]=(colorCounter % MAX_NO_COLORS);    // in case # tissue
    }
    > # colors
} // end for j
} // end sortByTissue()
//*****

public void sortByStage() // if "stage" in Choice List picked
    // sort libraries based on developmental stage
//*****

{

```

```

System.arraycopy(libraryNamesByStage,0,sortedLibraryNames,0,numberOfLibraries);
System.arraycopy(estCountByStage,0,estCountBySortedLib,0,numberOfLibraries);

// create string for text area
// establish colors for library names

librarySortString = "Sort by stage:\n";

for (int j = 0; j < numberOfLibraries; j++)
{
    if ( (j==0) ||
        ( !sortedStage[j].equals(sortedStage[j-1]) )
        )
    {
        librarySortString = librarySortString + "\n" + sortedStage[j] + ": "
            + sortedLibraryNames[j].trim();
        colorCounter++;          // change to new color
    }
    else // library from same stage
    {
        librarySortString = librarySortString + ",
"+sortedLibraryNames[j].trim();
    }
    // stage same->same color; stage different->different color
    sortColorIndex[j]=(colorCounter % MAX_NO_COLORS);    // in case # stage >
# colors

} // end for j
} // end sortByStage()
//*****

public void sortByCondition()    // if "condition" in Choice List picked
    // sort libraries based on treatment/condition
//*****

{

System.arraycopy(libraryNamesByCondition,0,sortedLibraryNames,0,numberOfLibrarie
s);

System.arraycopy(estCountByCondition,0,estCountBySortedLib,0,numberOfLibraries);

// create string for text area
// establish colors for library names

librarySortString = "Sort by condition:\n";

for (int j = 0; j < numberOfLibraries; j++)
{
    if ( (j==0) ||
        ( !sortedCondition[j].equals(sortedCondition[j-1]) )
        )
    {
        librarySortString = librarySortString + "\n" + sortedCondition[j] + ": "
            + sortedLibraryNames[j].trim();
        colorCounter++;          // change to new color
    }

```

```

    }
    else // library from same condition
    {
        librarySortString = librarySortString + ",
"+sortedLibraryNames[j].trim();
    }
    // condition same->same color; condition different->different color
    sortColorIndex[j]=(colorCounter % MAX_NO_COLORS); // in case #
condition > # colors
    } // end for j
    } // end sortByCondition()
//*****

    public int getXYContig(String contig) // find x,y for contig sought
(specified in textfield)
//*****

    {
        // called by highlightThisContig()
        int i = 0;
        int indexToShow = -1;
        boolean contigNotFound = true;

        while ( (contigNotFound) && (i < numberOfContigs) )
        {
            if ( contigNames[i].equals(contig) )
            {
                contigNotFound = false;
                indexToShow = i;
            }
            i++;
        } // end while

        return indexToShow;
    } //end getXYContig()
//*****

    public void showContigsFromLibrary1() // user has selected lib name -> show
associated contigs
//*****

    {
        // called by whichContigsFromLib1.ItemListener
        boolean libraryFound=false;

        // HIGHLIGHT ALL CONTIGS WITH ESTS FROM THIS LIBRARY
        try
        {
            if ( (!whichContigsFromLib1.getSelectedItem().equals("      None
")) &&

(!whichContigsFromLib1.getSelectedItem().equals(libraryRequested1.trim())) &&
                // prevent dupliC processing

(!whichContigsFromLib1.getSelectedItem().equals(whichContigsFromLib2.getSelected
Item())) &&

(!whichContigsFromLib1.getSelectedItem().equals(whichContigsFromLib3.getSelected
Item())) ) // prevent dupliC processing
        {

```

```

        showContigsInLib1 = true;

        libraryRequested1 = whichContigsFromLib1.getSelectedItemAt(); // choice list
selection

        int i=0;

        while ( (!libraryFound) && (i<numberOfLibraries) )
        {
            if ( libraryRequested1.equals(sortedLibraryNames[i]) )
            {
                indexRequestedLib1 = i;
                libraryFound=true;
            }
            i++;
        }
        contigsInLibrary1=getContigsForLibrary(libraryRequested1);
        showStatus("highlighting contigs with ESTs from
"+libraryRequested1.trim());
    }
    else if ( whichContigsFromLib1.getSelectedItemAt().equals("      None
") ) )
    {
        libraryRequested1 = ""; // so that cancelled highlight-selection not
retained
        showContigsInLib1 = false;
    }
    else if (
(whichContigsFromLib1.getSelectedItemAt().equals(whichContigsFromLib2.getSelectedI
tem())) ||

(whichContigsFromLib1.getSelectedItemAt().equals(whichContigsFromLib3.getSelectedI
tem())) )
        // or asking for same library as in another choice list
        {
            libraryRequested1 = ""; // so that cancelled highlight-selection not
retained
            showContigsInLib1 = false;
            whichContigsFromLib1.select("      None      ");
        } // end if
    }
    catch (Exception e)
    {
        err1+="showContigsFromL1: "+e;
    }

    repaint();
} // end showContigsFromLibrary1()
//*****
public void showContigsFromLibrary2() // user has selected lib name -> show
associated contigs
//*****

{
    boolean libraryFound=false;

    // HIGHLIGHT ALL CONTIGS WITH ESTS FROM THIS LIBRARY

```

```

        if ( (!whichContigsFromLib2.getSelectedItem().equals("          None
")) &&

(!whichContigsFromLib2.getSelectedItem().equals(libraryRequested2.trim())) &&
        // prevent duplc processing

(!whichContigsFromLib2.getSelectedItem().equals(whichContigsFromLib1.getSelected
Item())) &&

(!whichContigsFromLib2.getSelectedItem().equals(whichContigsFromLib3.getSelected
Item())) ) // prevent duplc processing
    {
        showContigsInLib2 = true;

        libraryRequested2 = whichContigsFromLib2.getSelectedItem(); // choice list
selection

        int i=0;

        while ( (!libraryFound) && (i<numberOfLibraries) )
        {
            if ( libraryRequested2.equals(sortedLibraryNames[i]) )
            {
                indexRequestedLib2 = i;
                libraryFound=true;
            }
            i++;
        }

        contigsInLibrary2=getContigsForLibrary(libraryRequested2);
        showStatus("highlighting contigs with ESTs from
"+libraryRequested2.trim());
    }
    else if ( whichContigsFromLib2.getSelectedItem().equals("          None
") )
    {
        libraryRequested2 = ""; // so that contigs not selected for subseq lib
selection
        showContigsInLib2 = false;
    }
    else if (
(whichContigsFromLib2.getSelectedItem().equals(whichContigsFromLib1.getSelectedI
tem())) ||

(whichContigsFromLib2.getSelectedItem().equals(whichContigsFromLib3.getSelectedI
tem())) )
        // or asking for same library as in another choice list
        {
            libraryRequested2 = ""; // so that contigs not selected for subseq lib
selection
            showContigsInLib2 = false;
            whichContigsFromLib2.select("          None          ");
        } // end if
        repaint();
    } // end showContigsFromLibrary2()
//*****

```

```

    public void showContigsFromLibrary3()    // user has selected lib name -> show
associated contigs
//*****

{
    boolean libraryFound=false;

    // HIGHLIGHT ALL CONTIGS WITH ESTS FROM THIS LIBRARY

    if ( (!whichContigsFromLib3.getSelectedItem().equals("        None
")) &&

(!whichContigsFromLib3.getSelectedItem().equals(libraryRequested3.trim())) &&
    // prevent duplec processing

(!whichContigsFromLib3.getSelectedItem().equals(whichContigsFromLib1.getSelected
Item())) &&

(!whichContigsFromLib3.getSelectedItem().equals(whichContigsFromLib2.getSelected
Item())) ) // prevent duplec processing
    {
        showContigsInLib3 = true;

        libraryRequested3 = whichContigsFromLib3.getSelectedItem();

        int i=0;

        while ( (!libraryFound) && (i<numberOfLibraries) )
        {
            if ( libraryRequested3.equals(sortedLibraryNames[i]) )
            {
                indexRequestedLib3 = i;
                libraryFound=true;
            }
            i++;
        }
        contigsInLibrary3=getContigsForLibrary(libraryRequested3);
        showStatus("highlighting contigs with ESTs from
"+libraryRequested3.trim());
    }
    else if ( whichContigsFromLib3.getSelectedItem().equals("        None
") )
    {
        libraryRequested3 = "";
        showContigsInLib3 = false;
    }
    else if (
(whichContigsFromLib3.getSelectedItem().equals(whichContigsFromLib1.getSelectedI
tem())) ||

(whichContigsFromLib3.getSelectedItem().equals(whichContigsFromLib2.getSelectedI
tem())) )
        // or asking for same library as in another choice list
        {
            libraryRequested3 = "";
            showContigsInLib3 = false;
            whichContigsFromLib3.select("        None        ");

```

```

    } // end if

    repaint();
} // end showContigsFromLibrary3()
//*****
public void highlightTheseContigs()    // user has entered contig#->pinpoint
where contig is
//*****

{
    try
    {
        iContigs = new int[numInContigList];    // array of indices to x,y
coordinates

        int i=0;

        while(st.hasMoreTokens())
        {
            if (i > 0)    // not the first contig
            {
                iContigs[i++]=getXYContig("C_Contig"+st.nextToken());    // put
contig in an array
            }
            else    // the first contig
            {
                // draw library lines for first contig in list
                contigNumber=st.nextToken();    // trim() not needed
                contigToShow = "C_Contig"+contigNumber;    // grab text from
textfield
                iContigs[i++] = iContig = getXYContig(contigToShow);    // get index
for contig in contigNames[] (& in x,y arrays)
            }
        } // end while
    }
    catch (Exception e)
    {
        err1+="highlightTheseContigs: "+e;
    }
} // end highlightTheseContigs()
//*****
public void showBestHit()    // bring up window with page showing best BLAST
alignment for contig
//*****

{
    try
    {
        // will have to be revised once we get contigs from 2nd assembly loaded.
        // "windowName" instead of _blank doesn't change window title
        // NB: URL will not work in appletviewer, so comment out showBestHit()
when testing in appletviewer
        getAppletContext().showDocument(new URL("http://wheat.pw.usda.gov/cgi-
bin/westsql/contig.cgi?q=NSFT01P2_Contig"+contigNumber+"&i=e&t=c"), "_blank");
    }
    catch (MalformedURLException e)
    {

```

```

        showStatus("URL not found for Contig "+contigNumber);    // moot as
Hummel's page has error message
    }
} // end showBestHit()
//*****
    public void showBestHits() // bring up window with page showing best BLAST
alignment for contig
//*****

    {
        try
        {
            String contigToken;
            while(st.hasMoreTokens())
            {
                contigToken = st.nextToken();
                // will have to be revised once we get contigs from 2nd assembly
loaded:
                getAppletContext().showDocument(new URL("http://wheat.pw.usda.gov/cgi-
bin/westsql/contig.cgi?q=NSFT01P2_Contig"+contigToken+"&i=e&t=c"), "_blank");
                // "windowName" instead of _blank doesn't change window title
            } // end while
        }
        catch(MalformedURLException e)
        {
            showStatus("URL not found for "+st.nextToken());    // moot as Hummel's
page has error message
        }
    } // end showBestHits()
//*****
    public void sortLibraries() // user has selected sort criterion in choice
list->redraw libraries/contigs
//*****

    {
        clickMouseX=0;                // click point irrelevant after sort
        drawRectAroundContigs = false;
        getListOfContigs = false;
        colorCounter = -1;

        try
        {
            //LIBRARY SORT ORDER
            if ( !libraryOrder.getSelectedItem().equals(" CUSTOM") )    // not custom
sort
            {
                if ( (libraryOrder.getSelectedItem().equals(" germplasm")) &&
(!sortAlready.equals("germplasm")) )    // don't redo the same
sort
                {
                    sortAlready="germplasm";
                    sortByGermplasm();
                    sortTitle.setText("Sort will take up to 45 seconds");    // to alert
the user
                    calculateSortedLibContribution();
                    highlightCorrectLib();
                }
            }
        }
    }

```

```

else if ( (libraryOrder.getSelectedItem().equals(" tissue")) &&
          (!sortAlready.equals("tissue"))) )
{
    sortAlready="tissue";
    sortByTissue();
    sortTitle.setText("Sort will take up to 45 seconds");
    calculateSortedLibContribution();
    highlightCorrectLib();
}
else if ( (libraryOrder.getSelectedItem().equals(" dev_stage")) &&
          (!sortAlready.equals("dev_stage"))) )
{
    sortAlready="dev_stage";
    sortByStage();
    sortTitle.setText("Sort will take up to 45 seconds");
    calculateSortedLibContribution();
    highlightCorrectLib();
}
else if ( (libraryOrder.getSelectedItem().equals(" condition")) &&
          (!sortAlready.equals("condition"))) )
{
    sortAlready="condition";
    sortByCondition();
    sortTitle.setText("Sort will take up to 45 seconds");
    calculateSortedLibContribution();
    highlightCorrectLib();
}
else if ( (libraryOrder.getSelectedItem().equals(" species")) &&
          (!sortAlready.equals("species"))) )
{
    sortAlready="species";
    sortBySpecies();
    sortTitle.setText("Sort will take up to 45 seconds");
    calculateSortedLibContribution();
    highlightCorrectLib();
}
else // user has chosen the same sort category as before
{
    sortTitle.setText("Choose another sort category");          // user needs
to pick something else
}
}
else // CUSTOM
{
    sortAlready="custom";
    frCustom1.setVisible(true); // bring up new window with choice box to
select sort category
}
}
catch (Exception e)
{
    err1+="sortLibraries(): "+e;
}
} // end sortLibraries()
//*****
public void highlightCorrectLib()

```

```

//*****
{
    try
    {
        // make sure correct library name highlighted after sort
        for (int i = 0; i< numberOfLibraries; i++)
        {
            // library #1
            if (
whichContigsFromLib1.getSelectedItemAt().equals(sortedLibraryNames[i].trim()) )
            {
                indexRequestedLib1 = i;
                continue;          // skip rest of loop and start next iteration
            }

            // library #2
            if (
whichContigsFromLib2.getSelectedItemAt().equals(sortedLibraryNames[i].trim()) )
            {
                indexRequestedLib2 = i;
                continue;
            }

            // library #3
            if (
whichContigsFromLib3.getSelectedItemAt().equals(sortedLibraryNames[i].trim()) )
            {
                indexRequestedLib3 = i;
                continue;
            }
        } // end for

        sortTitle.setText("Select category to sort by");          // signals that sort
        completed

        repaint(); // required for sorted libraries to be displayed
    }
    catch (Exception e)
    {
        err1+="highlightCorrectLib error: "+e;
    }
} // end highlightCorrectLib()
//*****
public void customSortLibraries() // user has selected sort criterion in
choice list->redraw libraries/contigs
//*****

{
    colorCounter=-1;
    clickMouseX=0;          // click point irrelevant after sort
    drawRectAroundContigs = false;
    getListOfContigs = false;
    numberOfESTs=0;

    try
    {

```

```

        for (int i = 0; i < numberOfLibraries; i++)
        {
            sortedLibraryNames[i]="";          // reinitialize - unnec. for
estCountBySortedLib[]
        }

        // go through sortedCustom[] and populate sortedLibraryNames[] and
estCountBySortedLib[]
        for (int i = 0; i < numberOfLibraries; i++) // i - standard sort
        {
            for (int j=0; j< numberOfLibraries; j++)      // j - custom sort
            {
                if ( sortedCustom[j].equals(sortedStandard[i]) && // same sort feature
                    sortedLibraryNames[j].equals("") )           // if > 1 lib w/
same sort feature
                {
                    sortedLibraryNames[j] = libraryNamesByStandardSort[i];      //
populate arrays
                    estCountBySortedLib[j] = estCountByStandardSort[i];

                    numberOfESTs=numberOfESTs+estCountBySortedLib[j];              // for #
in upper left corner
                    break; // go back to outer "for" loop
                }
            } // end for j-sorted
        } // for i-unsorted

        // create string for text area and set library sort colors
        librarySortString = "Custom sort:\n";

        for (int j = 0; j < numberOfLibraries; j++)
        {
            if ( ( j==0 ) ||
                ( !sortedCustom[j].equals(sortedCustom[j-1]) ) // start a new row
            )
            {
                librarySortString = librarySortString + "\n" + sortedCustom[j] + ": "
                    + sortedLibraryNames[j].trim();
                colorCounter++;          // change to new color
            }
            else // library has same sort feature as previous library
            {
                librarySortString = librarySortString + ",
"+sortedLibraryNames[j].trim();
            } // end if

            // same feature->same color; different feature->different color
            sortColorIndex[j]=(colorCounter % MAX_NO_COLORS);    // in case #
features > # colors

        } // end for

        calculateSortedLibContribution();    // figure out which libraries
contributed to contig

        if ( !initialLoad )    // if not the first custom sort
        {

```

```

        highlightCorrectLib();
    }
    else // if ( initialLoad ), then OK to do rest of GUI with next repaint()
    {
        initialLoad = false;
        sortTitle.setText("Select category to sort by");    // signals that sort
completed
        repaint();
    } // end if
    } catch (Exception e)
    {
        err1=err1+"customSortLib: "+e;
    } // end try
    } // end customSortLibraries()
//*****
    public void fillSortedCustom()    // process sort feature selection
//*****

    {
        try
        {
            // populate sortedCustom[] with libraryCustomOrder-ordered items
            int i=-1;    // index for custom-sorted features
            for (int j=0; j< sortFeatureCount; j++)    // cycle through List of
(unique) sort features
            {
                for (int k=0; k< numberOfLibraries; k++)    // cycle through array of
non-unique sort features
                {
                    if ( libraryCustomOrder.getItem(j).equals(sortedStandard[k]) )    //
feature in unique list == feature in original
                    {
                        sortedCustom[++i] = libraryCustomOrder.getItem(j); // OK
                    } // end if
                } // end for k
            } // end for j
            sortTitle.setText("Sort will take up to 45 seconds");    // to alert
the user
            customSortLibraries();    // sort libraries to
match order of features
        }
        catch (Exception e)
        {
            err1=err1+"fillSortedCustom error: "+e;
        }
    } // end fillSortedCustom()
//*****
    public void doWhereAreContigsProcess() // textField action event or after
library sort
//*****

    {
        try
        {
            if ( !whereAreContigs.getText().equals("") )    // non-empty string
entered in contig list text area

```

```

        {
            lookingForContig = true;                // at least one contig sought
            st = new StringTokenizer(WhereAreContigs.getText()); // parse using
space, tab, \n or \r
            numInContigList = st.countTokens();      // count # contigs
requested
        }

        if ( numInContigList > 1)
        {
            lookingForContigs = true;                // multiple contigs requested
        }
        else
        {
            lookingForContigs = false;
        }
        highlightTheseContigs();                    // fcn call for at least one
contig
    }
    else // nothing entered in textfield
    {
        lookingForContigs = false;
        lookingForContig = false;
        numInContigList = 0;
    }
}
catch (Exception e)
{
    err1+="doWhereAreContigsProcess error: "+e;
}
} // end doWhereAreContigsProcess()
//*****
// end non-overridden / original functions, start overridden functions
//*****
    public void mouseEntered(MouseEvent me) {} // empty implementations instead of
    public void mouseExited(MouseEvent me) {} // using MouseAdapter class
    public void mouseDragged(MouseEvent me) {} // didn't implement box-drawing
with mouseDrag because repaint()->flashing
//*****

    public void mouseClicked(MouseEvent me)
//*****

    {
/**
    // save coordinates where mouse clicked
    clickMouseX = me.getX();
    clickMouseY = me.getY();

    if ( (clickMouseX >= START_X) &&
        (clickMouseX <= START_X+2*CIRCLE_RADIUS_INT) &&
        (clickMouseY >= START_Y) &&
        (clickMouseY <= START_Y+2*CIRCLE_RADIUS_INT) )

*/
//test
// impossible alternative: figure out how to designate circular boundary.
// if clicking in square area that surrounds circle:
    if ( (me.getX() >= START_X) &&

```

```

        (me.getX() <= START_X+2*CIRCLE_RADIUS_INT) &&
        (me.getY() >= START_Y) &&
        (me.getY() <= START_Y+2*CIRCLE_RADIUS_INT) )
    {
        // save coordinates where mouse clicked
        clickMouseX = me.getX();
        clickMouseY = me.getY();

        // find contig at this x,y coordinate
        clickedContig=findContigForXY(clickMouseX, clickMouseY);

        // so that representational mode retained when click event followed by
        another event
        if (showEqual)
        {
            clickedMode=" (E) ";
        }
        else if (showProportional)
        {
            clickedMode=" (P) ";
        }
        else // showWeighted
        {
            clickedMode=" (W) ";
        }

        repaint();
    } // end if
/**
//test -- to prevent contig label in wrong place if 2nd location clicked too
quickly
    else
    {
        clickMouseX=0;
    }
*/
} // end mouseClicked()
//*****
public void mousePressed(MouseEvent me)
//*****

{
    pressMouseX = me.getX();
    pressMouseY = me.getY();
} // end mousePressed()
//*****

public void mouseReleased(MouseEvent me)
//*****

{
    releaseMouseX = me.getX();
    releaseMouseY = me.getY();

    // make sure user started drawing inside (well, near) circle area
    if ( (pressMouseX <= START_X+2*CIRCLE_RADIUS_INT) &&
        (releaseMouseX >= START_X) &&

```

```

        (releaseMouseY >= START_Y) &&
        (pressMouseY <= START_Y+2*CIRCLE_RADIUS_INT ) &&
        ( ((pressMouseX >= START_X) && (releaseMouseX <=
START_X+2*CIRCLE_RADIUS_INT)) ||
        ((pressMouseY >= START_Y) && (releaseMouseY <=
START_Y+2*CIRCLE_RADIUS_INT))
    )
)
{
drawRectAroundContigs=true;
getListOfContigs=true;
contigListString = "";    // reset

if (showEqual)                // show Equal (default) library contribution
{
    for (int i=0; i<numberOfContigs; i++)
    {
        if ( (xCoordEContig[i] >= pressMouseX) &&
            (xCoordEContig[i] <= releaseMouseX) &&
            (yCoordEContig[i] >= pressMouseY) &&
            (yCoordEContig[i] <= releaseMouseY) )
        {
            contigListString = contigListString + contigNames[i].substring(8) + "
";
        }
    }
}
else if (showProportional)    // show Proportional
{
    for (int i=0; i<numberOfContigs; i++)
    {
        if ( (xCoordPContig[i] >= pressMouseX) &&
            (xCoordPContig[i] <= releaseMouseX) &&
            (yCoordPContig[i] >= pressMouseY) &&
            (yCoordPContig[i] <= releaseMouseY) )
        {
            contigListString = contigListString + contigNames[i].substring(8)+ "
";
            // strip over prefix, just print #
        }
    } // end for
}
else                            // show WEIGHTED
{
    for (int i=0; i<numberOfContigs; i++)
    {
        if ( (xCoordWContig[i] >= pressMouseX) &&
            (xCoordWContig[i] <= releaseMouseX) &&
            (yCoordWContig[i] >= pressMouseY) &&
            (yCoordWContig[i] <= releaseMouseY) )
        {
            contigListString = contigListString + contigNames[i].substring(8) + "
";
        }
    }
} // end if showEqual

if ( ( pressMouseX == releaseMouseX) &&

```

```

        ( pressMouseY == releaseMouseY) )           // press/release same point =
click point
{
    repaint(630,480,290,45);           // prevent drawing library lines two times
    // just update lower text area
to show clickedContig
}
else // mouse was dragged so that x,y changed
{
    repaint();           // update entire display area
}
} // end if user started drawing in circle area
//test
else
{
    drawRectAroundContigs=false;
    getListOfContigs=false;
}
} // end mouseReleased()
//*****

public void mouseMoved(MouseEvent me) // mouseover
//*****

{
    String mouseLocation = "";           // name of point that mouse is passing over

    moveMouseX = me.getX();
    moveMouseY = me.getY();

    mouseLocation="" + findContigForXY(moveMouseX,moveMouseY);

    if ( mouseLocation.equals("") ) // no contig found at this location
    // look for library information to display in status bar
    {
        for (int i=0; i<numberOfLibraries; i++)
        {
            if ( (
                (x[i] > xCircle)           &&
                (y[i] < yCircle)           && // lib coordinates in upper
right quadrant
                (x[i]+2 < moveMouseX )           &&
                (moveMouseX < x[i]+2+x_ADJUSTMENT) &&
                (y[i]-y_ADJUSTMENT-9 < moveMouseY) &&
                (moveMouseY < y[i]))
            || (
                (x[i] > xCircle)           &&
                (y[i] >= yCircle)           && // lower right quadrant
                (x[i]+2 < moveMouseX)           &&
                (moveMouseX < x[i]+2+x_ADJUSTMENT) &&
                (y[i] < moveMouseY)           &&
                (moveMouseY < y[i]+y_ADJUSTMENT+9))
            || (
                (x[i] <= xCircle)           &&
                (y[i] >= yCircle)           && // lower left quadrant
                (x[i]-x_ADJUSTMENT < moveMouseX) &&
                (moveMouseX < x[i])           &&
                (y[i] < moveMouseY)           &&

```

```

        (moveMouseY < y[i]+y_ADJUSTMENT+9))
    || (
        (x[i] <= xCircle) &&
        (y[i] < yCircle) &&
        (x[i]-x_ADJUSTMENT < moveMouseX)           &&    // upper left
quadrant
        (moveMouseX < x[i])           &&
        (y[i]-y_ADJUSTMENT-9 < moveMouseY)       &&
        (moveMouseY < y[i])) )
    {
        boolean libNotFound=true;
        int j=0;

        while ( (libNotFound) && (j<numberOfSortableLibs) )    // cycle thru
array of libraries w/sort features
        {
            if ( LibInfo[j].trim().equals(sortedLibraryNames[i].trim()) )
            {
                libNotFound=false;
                mouseLocation=LibInfo[j].trim()+" : "
                    +Germplasm[j].trim()+" "
                +Tissue[j].trim()+" "
                +Stage[j].trim()+" "
                +Condition[j].trim();
            }
            j++;
        } // end while
        break; // break out of for loop since library found
    } // end if
} // end for i
} // end if..else

        showStatus(""+mouseLocation);    // display contig or library in status bar
    } // end mouseMoved()
//*****

    public void itemStateChanged(ItemEvent ie)    // checkboxes & radio buttons
//*****
    {
        // clicked --> repaint
        // remove rectangle and list of contigs as obsolete once library
representation changed
        drawRectAroundContigs=false;
        getListOfContigs=false;

        // PROPORTIONAL OR EQUAL LIBRARY REPRESENTATION
        if ( Proportional.getState() ) // "Proportional" radio button selected
        {
            showProportional=true;
            showEqual=false;
            showWeighted=false;
        }
        else if ( Equal.getState() ) // "Equal" radio button selected
        {
            showProportional=false;
            showEqual=true;
            showWeighted=false;
        }
    }

```

```

else // "Weighted" radio button selected
{
    showProportional=false;
    showEqual=false;
    showWeighted=true;
} // end if

repaint(); // calls update(g) (default version of update(g)
calls paint(g))
} // end itemStateChanged
//*****

public void actionPerformed(ActionEvent ae) // buttons clicked, text in
textfield, then...
//*****

{
    // need to define actionPerformed() for CViewer class to compile
    // commands all moved out to individual event triggers
    // repaint();
    // repaint(left,top,width,height) to repaint specific part of window
} // end actionPerformed
//*****
// un-comment if I need textListener later...
// public void textValueChanged(TextEvent te)
// { // need to define textValueChanged() for CViewer class to compile
// }

//*****
// START INNER CLASS CViewerHelpFrame

//*****
class CViewerHelpFrame extends Frame
// pop-up window when Help button clicked
{
    CViewerHelpFrame(String title) // constructor
    {
        super(title); // call Frame's constructor

        //create object to handle window events, then register it to receive those
events
        addWindowListener(new WindowAdapter() // addWL
defines/instantiates anon inner class
        { // START anon inner class
            public void windowClosing(WindowEvent we)
            {
                setVisible(false); // removes window from screen when closed
            } // END anon inner class
        });
        // set Background needs to be in constructor, not init/paint() to work
        setBackground(WHITE_YELLOW);
    } // end constructor

    //init() not allowed in Frame

//*****

```

```

// paint() for CViewerHelpFrame
public void paint (Graphics g) // text displayed in help box
{
    g.drawString("LIBRARIES",10,70);
    g.drawString("- are displayed on the perimeter of the circle.",10,82);
    g.drawString("- \"SET LIBRARIES\" button:",10,94);
    g.drawString(" = to choose a different set of libraries.",10,106);
    g.drawString(" = to check the germplasm, developmental stage, tissue, or
condition for a library.", 10,118);
    g.drawString("- \"SHOW CONTIGS IN LIBRARY\" choice lists:",10,130);
    g.drawString(" = to highlight contigs with ESTs coming from a specific
library.",10,142);
    g.drawString("- \"SELECT CATEGORY TO SORT BY\" choice list:",10,154);
    g.drawString(" = to sort libraries by species, germplasm, developmental
stage, tissue,",10,166);
    g.drawString(" or condition, with sort features in alphanumeric order
within the category.",10,178);
    g.drawString(" = a CUSTOM sort will put the features within a category in
user-specified order.",10,190);
    g.drawString(" = sort will take up to 45 seconds.",10,202);
    g.drawString(" = will display libraries by sort feature around the circle
and in the middle text area.",10,214);

    g.drawString("CONTIGS",10,234);
    g.drawString("- are displayed as points inside the circle.",10,246);
    g.drawString("- contig positions reflect \"pull\" of libraries associated
with contig's component ESTs.",10,258);
    g.drawString("- Radio buttons to set library representation
mode:",10,270);
    g.drawString(" = PROPORTIONAL: contig's proximity to a library
proportional to # ESTs from library.",10,282);
    g.drawString(" = EQUAL: each contributing library has equal
\"pull.\".",10,294);
    g.drawString(" = WEIGHTED: accounts for #ESTs from library in this
contig as well as in all contigs.",10,306);
    g.drawString("- \"ENTER CONTIG #(S)\" textfield:",10,318);
    g.drawString(" = to find location of one or more contigs, enter numbers
separated by whitespace",10,330);
    g.drawString(" (e.g., '123 4567' for C_Contig123. &
C_Contig4567).",10,342);
    g.drawString(" = Lines will be drawn from the first contig to
contributing libraries.",10,354);
    g.drawString(" -> the number of ESTs from each library in the contig
will be shown",10,366);
    g.drawString(" => next to the lines (PROPORTIONAL and WEIGHTED
modes).",10,378);
    g.drawString(" => next to the library names (PROPORTIONAL and
EQUAL modes).",10,390);
    g.drawString(" -> the total ESTs--in all contigs--from a library will
be shown next to",10,402);
    g.drawString(" the library names (WEIGHTED mode).",10,414);
    g.drawString(" = contigs' component ESTs & libraries will be displayed
in the upper text area.",10,426);
    g.drawString(" = \"GET BEST HIT(S)\" button:",10,438);
    g.drawString(" -> to get best BLAST alignment and other info on
contigs entered in the textfield.",10,450);
    g.drawString("- To get information on a dot in the circle:",10,462);

```

```

        g.drawString("    = Mouse over dot until you see a contig name in the
status bar at bottom of browser.",10,474);
        g.drawString("    = Then click on dot (or enter contig number in \"Enter
contig #(s)\" textfield).",10,486);
        g.drawString("    = Lines to contributing libraries and component ESTs &
libraries will be displayed.",10,498);
        g.drawString("- To find out which contigs are represented by a cluster of
dots:",10,510);
        g.drawString("    = Draw a box around the contigs (see instructions in
lower text area).",10,522);
        g.drawString("    = You will see the contig numbers for this cluster in the
lower text area.",10,534);
    } // end paint CViewerHelpFrame
} // end class CViewerHelpFrame

//*****

// END INNER CLASS CViewerHelpFrame

//*****

//*****

// START INNER CLASS CViewerCustomCategoryFrameOne

//*****
class CViewerCustomCategoryFrameOne extends Frame // choose
// pop-up window when 'CUSTOM' sort category selected
{
    CViewerCustomCategoryFrameOne(String title) // constructor
    {
        super(title); // call Frame's constructor

        //create object to handle window events & register it to receive those
events
        addWindowListener(new WindowAdapter() // addWL defines/instantiates
anon inner class
        {
            // START anon inner class
            public void windowClosing(WindowEvent we)
            {
                setVisible(false); // removes window from screen when
closed
            } // END anon inner class
        });
        setLayout(null);
        setBackground(WHITE_YELLOW);
        // defining choice list and adding items to choice list in constructor
doesn't work
    } // end constructor

//*****

// paint() for CViewerCustomCategoryFrameOne
public void paint(Graphics g) // for CViewerCustomCategoryFrameOne
{

```

```

        g.drawString("Choose category to sort by: ",66,60);
    }

    public void stop()
    {
        setVisible(false);
    }

} // end class CViewerCustomCategoryFrameOne

//*****

// END INNER CLASS CViewerCustomCategoryFrameOne

//*****

//*****

// START INNER CLASS CViewerCustomFeatureFrameTwo (applet can't see if put
inside FrameOne)

//*****

class CViewerCustomFeatureFrameTwo extends Frame
// pop-up window when selection made in CViewerCustomCategoryFrameOne
{
    CViewerCustomFeatureFrameTwo(String title) // constructor
    {
        super(title); // call Frame's constructor

        //create an object to handle window evetns & register it to receive those
events
        addWindowListener(new WindowAdapter() // addWL creates anonymous inner
class
        {
            // START inner class
            public void windowClosing(WindowEvent we)
            {
                setVisible(false); // removes window from screen when
closed
            } // END inner class
        });
        setLayout(null); // kill layout manager --> manually-
set layout
        setBackground(WHITE_YELLOW);
    } // end constructor

//*****

// paint() for CViewerCustomFeatureFrameTwo
public void paint(Graphics g) // for CViewerCustomFeatureFrameTwo
{
    g.drawString("Select feature, then click 'Move Up'",250,100);
    g.drawString("Sort will take up to 45 seconds",252,265);
}
public void stop()

```

```

        {
            setVisible(false);
        }

    } // end class CViewerCustomFeatureFrameTwo

//*****

    // END INNER CLASS CViewerCustomFeatureFrameTwo

//*****

//*****

    // START INNER CLASS CViewerChooseLibFrame (user selects libraries to be
    displayed in applet)

//*****

    class CViewerChooseLibFrame extends Frame
    // pop-up window when URL for applet requested
    {
        CViewerChooseLibFrame(String title)          // constructor
        {
            super(title);                          // call Frame's constructor

            //create an object to handle window events & register it to receive those
            events
            addWindowListener(new WindowAdapter()    // addWL creates anonymous inner
            class
            {
                // START inner class
                public void windowClosing(WindowEvent we)
                {
                    setVisible(false);              // removes window from screen when
                    closed
                }
            } // END inner class
        });
        setLayout(null);                          // kill layout manager --> manually-
        set layout
        setBackground(WHITE_YELLOW);
    } // end constructor

//*****

    // paint() for CViewerChooseLibFrame
    public void paint(Graphics g)                  // for CViewerChooseLibFrame
    {
        g.drawString("Select libraries you wish to view from AVAILABLE
list:", 430, 68);
        g.drawString("REFERENCE", 28, 94);
        g.drawString("Up to 50 seconds", 785, 495);
        g.drawString("to process selection", 785, 510);
        g.setColor(DARK_BROWN);
        g.drawString("AVAILABLE", 428, 94);
    }

```

```

        g.setColor(MEDIUM_GREEN);
        g.drawString("SELECTED",663,94);
    }
    public void stop()
    {
        setVisible(false);
    }

} // end class CViewerChooseLibFrame

//*****

// END INNER CLASS CViewerChooseLibFrame

//*****

} // close class CViewer
//*****
// END CLASS CViewer
//*****
// END OF PROGRAM
//*****

```